VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ
DEPARTMENT OF COMPUTER SYSTEMS

DETEKTOR VIRŮ NA USB KLÍČENKÁCH POMOCÍ RASPBERRYPI

USB FLASHDRIVES VIRUS DETECTOR IMPLEMENTED IN RASPBERRYPI

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — **DOMINIK POLEHŇA**
AUTHOR

VEDOUCÍ PRÁCE — **Ing. MARTIN KRČMA**
SUPERVISOR

BRNO 2018

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačových systémů Akademický rok 2017/2018

Zadání bakalářské práce

Řešitel: **Polehňa Dominik**
Obor: Informační technologie
Téma: **Detektor virů na USB klíčenkách pomocí RaspberryPi**
USB Flashdrives Virus Detector Implemented in RaspberryPi
Kategorie: Bezpečnost

Pokyny:

1. Seznamte se s platformou RaspberryPi, jeho architekturou, vlastnostmi, způsoby použití a možnostmi programování.
2. Seznamte se se současnými antivirovými a antimalware systémy pro Linux.
3. Navrhněte systém založený na RaspberryPi pro automatickou detekci malware na USB discích.
4. Navžený systém implementujte.
5. Zhodnoťte vytvořený systém a navhněte další rozvoj.

Literatura:

- Dle doporučení vedoucícho.

Pro udělení zápočtu za první semestr je požadováno:

- Splnění bodů 1 až 3 zadání.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese http://www.fit.vutbr.cz/info/szz/

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Krčma Martin, Ing.**, UPSY FIT VUT
Datum zadání: 1. listopadu 2017
Datum odevzdání: 16. května 2018

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačových systémů a sítí
612 66 Brno, Božetěchova 2

L.S.

prof. Ing. Lukáš Sekanina, Ph.D.
vedoucí ústavu

Abstrakt

Práce je zaměřena na analýzu tématu okolo bezpečnosti na internetu a vytvoření detektoru virů na USB klíčenkách. Postupně jsou rozebírány a lehce nastíněny principy virů a antivirů. Ke zrealizování práce je využita platforma Raspberry Pi, jazyk Python a dostupné antivirové programy. Cílem je vytvoření automatického detektoru, který nepotřebuje interakci s uživatelem ke svému chodu.

Abstract

This thesis is focused on the analysis of internet security and the implementation of USB flashdrives virus detector. We will firstly analyze the basics of viruses and antiviruses and from gained knowledge we are going to create an automatic virus detector which doesn't need an user intervention. For impelementation will be used a platform Raspberry Pi and programming language Python.

Klíčová slova

Raspberry Pi, škodlivý software, antivirové programy, Python

Keywords

Raspberry Pi, malicious software, antivirus software, Python

Citace

POLEHŇA, Dominik. *Detektor virů na USB klíčenkách pomocí RaspberryPi.* Brno, 2018. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Martin Krčma

Detektor virů na USB klíčenkách pomocí RaspberryPi

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Martina Krčmy. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Dominik Polehňa
9. května 2018

Poděkování

Tímto děkuji vedoucímu práci za poskytnutí odborné pomoci při vypracovávání práce a všem co se podíleli na vyřizování žádosti pro poskytnutí prostředků.

Obsah

Kapitola 1

Úvod

V dnešní době dochází k elektrifikaci většiny zařízení, které lidé denně používají při běžném životě. Málo které zařízení dnes nemá možnost připojení k internetové síti, které uživatelé vyžadují pro větší pohodlí, aby mohli svá zařízení mít různými způsoby propojeny nebo je vzdáleně monitorovat či obsluhovat. Mezi takové běžné zařízení s internetovým připojením patří v dnešní době osobní počítač, mobilní telefon, televize, ale pomalu více se začínají objevovat například i chytré domácnosti, automobily, lékařské zařízení apod. S příchodem takových zařízení, která se dají centrálně ovládat, přichází hrozba zneužití lidmi se špatnými úmysly, a to nelze brát na lehkou váhu. Na druhou stranu lze těchto vymožeností využít i ve prospěch lidstva. Pro představu lze uvést případ kdy, když v roce 2017 sužoval Spojené státy americké hurikán Irma, automobilka Tesla centrálně navýšila kapacitu baterií svých automobilů, aby lidé měli větší dojezd a stihli se tak dostat do bezpečí. Dokázat něco takového je pro svět velkým pokrokem, ale představa o tom, že má někdo plnou kontrolu nad cizím vozidlem, se může zdát z jistých úhlů pohledu právě poněkud nebezpečná.

Mnoho běžných uživatelů si myslí, že svá data mají na svém počítači v bezpečí. Avšak veškerá taková zařízení mohou být napadnutelná z internetové sítě pomocí různých škodlivých softwarů, kterých se dnes vyskytuje nespočetně mnoho a bránit se jim mnohdy není úplně jednoduché. Za tímhle účelem vzniklo v průběhu vývoje internetové sítě nespočetné množství firem, které se zabývají zabezpečováním zařízení.

Postupně v téhle práci bude rozebráno, co takový škodlivý software představuje, jaké existují jeho druhy, ale také programy, které uživatele brání před těmito škodlivými softwary. Ze všech nabytých vědomostí nakonec bude implementován automatický detektor virů, který bude využívat dostupné programy pro odhalování škodlivého softwaru. Celá aplikace bude běžet na platformě RaspberryPi, které bude věnována zvláštní kapitola.

Kapitola 2

Škodlivý software

Na úvod bude vysvětleno, co vůbec takový škodlivý software znamená. Může se vyskytovat také pod slovem malware, které vychází z anglického výrazu *malicious software*, červ, počítačový virus a mnoho dalších názvosloví. V práci se dále budou vyskytovat všechny uvedené pojmy, ale ve skutečnosti každý označuje jiný typ škodlivého softwaru. Nejčastěji se pro pojmenování škodlivého softwaru obecně používá slovo virus. Jedná se o poměrně zaužívanou věc a v práci se obecné pojmenování slovem virus může vyskytnout. Zařazení škodlivého softwaru do správné skupiny může být někdy problém, protože jejich chování často bývají variabilní a existují různá rozdělení. Následující sekce odrážejí jedno z možných rozdělení.

O počítačových virech se mluvilo přibližně již v 60. letech, kde byly pouze smyšlenou fantazií různých spisovatelů. Až v průběhu roku 1983 se podařilo autoru Dr. Frederikovi Cohenovi napsat samomnožicí se program. Tenhle program byl poprvé označen za počítačový virus, přestože se však jednalo pouze o neškodný kus kódu, který se dokázal samovolně šířit dál na zařízení, na kterém se vyskytoval. Jako první počítačový virus, který způsobil uživatelům nějakou škodu, se uvádí virus Brain, který byl sestrojen bratry Basid a Faaroq Alvi v roce 1986. Tím bylo dáno za vznik éře virů, která se průběžně vyvíjí do dnešní doby. Následuje rozdělení škodlivého softwaru, kde v každé sekci bude rozebráno, jaký způsob používají k proniknutí do zařízení a jejich nejčastější způsoby napadení.[50] [27]

2.1 Virus

V odstavci výše již bylo uvedeno, že hlavním znakem viru je jeho samošířitelnost. Pro své šíření je však zapotřebí nějakého hostitele. K tomu využívá různé spustitelné soubory. Pokud je takový infikovaný soubor spuštěn za pomocí uživatele, tak zároveň dojde ke spuštění viru, který se v souboru ukrýval. Při svém spuštění je pak virus schopný infikovat řadu dalších souborů, které opět při svém spuštění provedou infikaci. Tím si virus zajišťuje masivní šíření a složitější odstranění ze systému.[49]

Viry lze dále rozdělit na **nerezidentní** a **rezidentní**. **Nerezidentní** viry pracují takovým způsobem, že v okamžiku, kdy je spuštěn infikovaný soubor, přeberou řízení nad spuštěním, provedou svoji činnost, čímž nejčastěji dojde k infikaci dalších souborů a předají řízení zpátky spuštěnému programu. Tím docílí toho, že se vykoná uživatelův požadavek a ten tak nezjistí, že k nějaké infikaci souborů vlastně došlo. Druhou skupinou jsou viry **rezidentní**, které mají schopnost přetrvat rezidentně neboli skrytě, v paměti. Rezidentnost

programu tedy znamená, umožnění jeho běhu na pozadí operačního systému. Rozdělení do těchto dvou skupin se odráží podle umístění viru v paměti.

Dále lze viry dělit podle toho, jaký mají cíl útoku. To dělí viry na **bootovací**(zaváděcí) a **souborové**. Zde se vyskytují pojmy zaváděcí sektor (boot sector) a tabulka rozdělení pevného disku (partition table). Zaváděcí sektor obsahuje kód, který je proveden při zavedení operačního systému. Partition table slouží k rozdělení fyzického disku na jednotlivé oddíly. Bootovací virus uloží své tělo do zaváděcího sektoru nebo tabulky rozdělení. Infikování počítače je poté možné, při pokusu zavedení systému z fyzického disku, který má infikován buď zaváděcí sektor nebo tabulku rozdělení pevného disku. Oba případy vedou ke stejnému výsledku. Rozdíl je pouze z pohledu programátora viru. Druhá skupina, kterou jsou souborové viry, již z názvu říká, že napadají spustitelné soubory operačního systému. Jedná se o nejrozšířenější skupinu počítačových virů. Mezi standardní napadené soubory spadají soubory s příponami EXE, COM, BAT, OVL a BIN. Viry však můžou být i kombinací bootovacího a souborového viru. Viry tohoto typu lze označit například jako **multiparitní**.[13]

Jako poslední možností rozdělení virů, které zde bude uvedeno, je podle koncepce návrhu a projevu chování. První takovou skupinou jsou **stealth** viry. Slovíčko „stealth“ pochází z angličtiny a v překladu znamená „tajný“. To je hlavní vlastnost, kterou se viry ve skupině řídí. Určitým způsobem dochází ke skrývání změn, které byly provedeny v operačním systému. Toho lze docílit například tím, že při přepisování systémových souborů dojde k uložení jeho původní verze. Pokud později nastane situace, že operační systém bude daný systémový soubor požadovat, dojde k podstrčení původní verze souboru, a tak dojde k zakrytí změn před operačním systémem.[32] Stejný koncept tajnosti používají i viry **polymofrní**.[30] Nepracují však způsobem jako stealth viry, které reagují na systémové činnosti, ale využívají polymorfnosti. To znamená, že žádná těla z kopií nejsou totožná. Tím se brání detekci jejich přítomnosti v infikovaném souboru či sektoru. Poslední skupinou jsou **tunelující** viry, které se vyznačují tím, že své zápisy na disk neprovádí běžnou cestou, ale dojde k protunelování řetězců ovladačů zařízení v paměti. Na konec tohoto řetězce připojí své tělo a dále pak ovládají řadič pevného disku.

2.2 Makroviry

Dalším škodlivým softwarem jsou makroviry. Jedná se o skupinu virů, které se můžou vyskytovat v textových souborech. Tohle je především spjato s Microsoft Office soubory, jako jsou DOC, DOCX, XLS, PPT apod. Virus ke svému spuštění potřebuje, aby počítač provedl nějakou část kódu, což se u normálních textových souborů úplně tak neděje, aby jich bylo možné použít k šíření virů. Makroviry proto využívají Office soubory, které ukládají makra do stejného dokumentu, jako je text, prezentace či tabulka. Při spuštění souboru s makrovirem dojde k infikování celého programu z balíčku Microsoft Office. Makrovir se pak dále šíří do všech souborů, které jsou vytvořeny pomocí daného programu. Makroviry můžou používat stejné techniky jako obyčejné viry.[28]

2.3 Trojské koně

Trojský kůň neboli trojan, nemá nic společného s virem. Jejich jediná společná vlastnost je, že se snaží uživateli uškodit, ale jinak se jedná o dva různé typy škodlivých softwarů. Název této skupiny je odvozen z dřevěného koně, který byl při válce s Řeky darován Trójí jako dar ke konci válečného období. Dřevěný kůň však nebyl pouhým darem, ale byla to

past, která stála za dobitím Tróje. Uvnitř byli ukrytí řečtí vojáci. Na obdobné přetvářce jsou postaveny i počítačové trojské koně.

Trojské koně na rozdíl od virů nejsou schopny samošířitelnosti. Nejčastěji se skrývají pod spustitelnými soubory s příponou EXE, které se na první pohled tváří přátelsky, ale ve skutečnosti neobsahují to, co uživatel očekává, ale pouze tělo trojského koně. Z toho již může být zřejmé, že trojský kůň nepoužívá ke svému šíření žádného hostitele a je možné se ho zbavit tím, že dojde k odstranění podvodného souboru, který se nachází na zařízení. Jejich šíření a spuštění je více založeno na zvědavosti a nepozornosti uživatele.

Trojani můžou mít různé funkce. Jednou z nejznámějších funkcí je otvírání portů, které poskytují útočníkům vzdálený přístup do uživatelova počítače. Dále je možné se setkat s takzvaným keyloggerem, který zaznamenává stisky jednotlivých kláves na klávesnici. Sesbíraná data poté za pomocí emailové služby přeposílá útočníkovi. Z takových dat je možné poté vyčíst přihlašovací údaje třeba k sociálním sítím, ale i bankovním účtům. Dále může například sledovat historii prohlížení na internetu, posílat nežádoucí zprávy, kterým se říká spam aj.[20] [31]

2.4 Červi

Opět se jedná o typ škodlivého softwaru, který nepoužívá ke svému šíření žádného hostitele. Na rozdíl od obyčejných virů, které se převážně šíří lokálně na jednom zařízení, se červi snaží rozšířit i na jiná zařízení. Dříve k tomu převážně využívali síťové prostředky, které byli po infikování počítače schopni ovládat a zneužít tedy k dalšímu šíření. V dnešní době se červi šíří převážně pomocí elektronické pošty, kde jsou posílány jako soubor v příloze. Ačkoliv je červ schopen se samovolně šířit, může k tomu využít i zvědavost uživatele, který může být například pomocí elektronické pošty vybízen ke stažení červa z přílohy.

Kromě toho, že dochází k šíření červů na další zařízení, je úkolem červa například čerpat dostupné systémové zdroje jako je čas procesoru, diskový prostor apod. Můžou také využívat modifikaci registrů systému, čímž dochází k zajištění k pozdějšímu znovu aktivování červa.[14] [52]

2.5 Ransomware

Ransomware je oproti ostatním škodlivým softwarům poměrně mladší. První škodlivý software tohoto typu se ale objevil již někdy na konci 80. let, ale později nedocházelo k tak častému využívání tohoto typu. Jeho výskyt byl převážně v Rusku. Název opět vychází z angličtiny. Kokrétně ze slovíček *ransom* a *software*. V překladu to znamená vyděračský software. V dnešní době se ransomware poměrně rozšiřuje a v roce 2013 se vyskytovalo přibližně 500 000 druhů ransomwaru. Letos již pravděpodobně bude dávno překonána hranice jednoho milionu.

Z anglického překladu vychází, že použití tohoto typu škodlivého softwaru se používá pro vydírání lidí. Nejčastěji jsou po obětech vyžadovány peníze. Dojde-li k infikování ransomwarem, s největší pravděpodobností také dojde k zablokování počítačového systému nebo k zašifrování dat na disku. Novější typy ransomwaru používají k šifrování dat symetrický klíč a fixní veřejný klíč, případně šifru RSA. Privátní klíč, pomocí kterého lze data rozšifrovat, tak zná pouze strůjce ransomwaru. Pokud dojde k zašifrování dat, je uživatel vyzván k zaplacení určité částky, aby došlo k dešifrování dat a nedošlo tak k jejich ztrátě. Nemusí se však nutně jednat o šifrování dat nebo blokování přístupu, ale například i k omezení

rychlosti zařízení apod. Nejčastěji se šíří jako červi nebo trojské koně. To znamená, že k infikování systému dojde převážně kvůli nepozornosti uživatele, který si do svého zařízení ransomware stáhne.[9] [16]

2.6 Rootkit

Jedná se o speciální případ škodlivého softwaru, který může být použit ve spojení se vším, co bylo uvedeno v předchozích sekcích. Pokud útočník vytváří škodlivý software, většinou chce, aby napáchal co nejvíce škody. K tomu mu může pomoci, pokud je těžce dohledatelné umístění, případně jeho běh, v operačním systému. Právě k tomu slouží rootkity, které se snaží zabránit tomu, aby byl škodlivý proces viditelný v seznamu běžících procesů, případně omezují čtení infikovaných souborů, aby nedošlo k jeho odhalení. Může však i měnit položky registrů systému nebo řídit síťové spojení a systémové služby.

Původně bylo názvosloví rootkit používáno pro sadu nástrojů, které umožňovaly získat administrátorská oprávnění v systémech na unixovém jádře. Jednalo se o běžné nástroje pro správu systému, které nahrazovaly ty původní. To však bylo jednoduše odhalitelné a metody se postupně změnily. V dnešní době je proto názvosloví rootkit používané pro škodlivé softwary, které se snaží skrýt své aktivity.[33][34]

2.7 Spyware

Spyware se na rozdíl od ostatních škodlivých softwarů snaží ublížit jinou cestou. Při jeho činnosti dochází ke shromažďování informací o uživateli bez jeho vědomí. Informace nejčastěji bývají odeslány útočníkovi, který je zneužije ve svůj prospěch. Případně může přebírat řízení zařízení opět bez vědomí uživatele. V nejvíce případech se využívá ke stopování a sledování uživatele na webu, kde jsou informace předány třetím stranám a ty je využívají k zobrazování reklam.[40]

Software se však nevyužívá pouze k nelegální špionáži uživatelů. Státy mají různé zákony ohledně sledování uživatelů a některé státy mají dokonce vlastní spyware. Nejznámější případ ohledně sledování uživatelů má na starosti Facebook, který si předává s ostatními stránkami informace o jejich návštěvách.

2.8 Adware

V případě Adwaru se nejedná úplně tak o škodlivý software, ale spíše o obtěžujicí pro uživatele. Automaticky generuje online reklamy, které se zobrazují například při instalaci softwaru a díky nim získávají vývojáři peníze. Reklamy se můžou vyskytovat v různých formách. Může to být jen klasický obrázek s reklamním nápisem, banner, reklama přes celou obrazovku, video nebo vyskakovací reklamy apod. Většina vývojářů, kteří nabízejí jejich programy zdarma ke stažení, používají právě adware, aby získali nějaký výdělek. Pokud je pak na trhu dostupná i placená verze takového programu, jedná se o verzi právě bez těchto reklam.[53]

Adware bývá zahrnut do škodlivých softwarů hlavně z důvodů, že zobrazování takových reklam bývá ve většině případů uživatelem nežádoucí. Nejčastěji se jedná o různá vyskakovací okna nebo okna, která se nedají zavřít. I tak se však většinou jedná o neškodnou reklamu, a tak se některé firmy soudí s vývojaři antivirových programů za to, že jim adware blokují.

Kapitola 3

Antivirové programy

Doposud byly rozebírány programy, které slouží k poškození dat, vydírání lidí, zobrazování reklam aj. Pokud existují programy tohoto typu, existují i programy, které mají opačný úkol. Tím je bránit zařízení před napadením malwarem a při jeho napadení ho odhalit a zneškodnit.

K takovému účelu se využívají antivirové programy neboli antiviry. Jejich hlavním úkolem je identifikovat a odstranit počítačové viry na zařízení. Nicméně s postupným rozšiřováním ostatních druhů škodlivých softwarů, začaly antivirové programy poskytovat rozšířenější ochranu zařízení. Novější programy již běžně poskytují i ochranu například proti nežádoucím reklamám ve webovém prohlížeči, ransomwaru, keyloggerům, zadním vrátkům, rootkitům, trojským koňům, červům, podvodným nástrojům, reklamním a špionážním softwarům atp. V dnešní době je takových antivirů na trhu k dostání mnoho a využívají nejrůznější techniky k tomu, aby dosáhli nejlepších výsledků při identifikaci malwaru a překonaly tak svou konkurenci. Nyní bude krátce nastíněna historie antivirových programů a způsoby, které antiviry využívají k identifikaci virů.[36][26]

3.1 Historie

O vytvoření prvního antivirového programu se pře více lidí. Avšak první dokumentované zneškodnění viru se datuje k roku 1987. Jednalo se o virus Vienna, který infikoval soubory a modifikoval je tak, že při dalším jejich spuštění došlo k restartu systému. Zneškodnění viru bylo provedeno Evropanem Berndem Fixem.[54] Ještě tentýž rok dvojice, která založila firmu G Data Software, vypustila na svět antivirus pro platformu Atari. Poslední verze tohoto antiviru byla vydána v roce 2004.[10] Zároveň se jménem John McAfee, zakladatel firmy McAfee, vytvořil první verzi antiviru VirusScan.[15] Ve stejném roce vydala i bývalá československá firma, dnes již pouze slovenská, svoji první verzi antiviru NOD, který se na trhu drží do dnešní doby s kvalitními výsledky.[8]

3.2 Principy identifikace virů

Antivirové programy pracují dvěma základními principy. Jedním z nich je hledání sekvence odpovídající definici nějakého počítačového viru, která je uložena k porovnání v databázi známých virů. Druhý z nich se snaží detekovat nežádoucí aktivity v operačním systému, podle kterých dokáže vyhodnotit, zdali je systém infikován.

Aby byla zajištěna funkčnost první metody, je zapotřebí, aby měl antivir přístup k virové databázi nebo se též může vyskytovat pod názvem slovník. Při prohledávání souborů na fyzickém disku dochází k porovnávání souborů s virovou databází. Celkově z toho vyplývá, že úspěšnost odhalení viru při použití této metody, se nejvíce odvíjí od aktuálnosti virové databáze, proto je zapotřebí, aby byla databáze co nejaktuálnější, a právě o to se starají výrobci antivirových programů. Pokud dojde k nalezení shody definice s virem, nabízí ve většině případů antiviry tři možnosti. První z nich nabízí možnost odstranění infikovaného souboru z počítače, nebo je možnost pokusit se infikovaný soubor opravit odstraněním známého kusu kódu viru ze souboru. Poslední možností je využití tzv. karantény, kde je infikovaný soubor přesunut a je mu tak zabráněno následného šíření. Ani jedna z možností však nemusí zaručit úplného odstranění viru ze zařízení, protože vir již může být rozšířen v jiných souborech, které se mu nepodařilo identifikovat. Jako možnost v samozřejmě nabízí i neprovedení žádné akce a ponechat tak soubor na původním místě. Toho může být využito, pokud uživatel ví, že antivir hlásí infikaci na souboru, který žádný virus neobsahuje.

Druhá z možností používá k detekci nežádoucích aktivit různé heuristiky. Takový způsob je naprosto odlišný od prvního a nespoléhá se na žádnou aktuálnost virové databáze. Jeho výhodou je, že dokáže odhalit i nejnovější viry. Naopak má i svou nevýhodu a tou je to, že často dává uživateli falešný poplach. Hodně záleží na tom, jakým způsobem funguje použitá heuristika a do jaké hloubky kontroluje aktivitu v systému.[50]

Existují však i jiné dostupné metody, které se velmi často využívají. Jednou z takových metod je například **sandbox**. Metoda pracuje takovým způsobem, že spuštěný program pustí v simulaci, která odděluje běžící procesy. Díky tomu omezuje programu přístup v počítači pouze na určité zdroje a tím má program zabráněno modifikovat cokoliv jiného. Po ukončení programu většinou dochází k analýze změn v sandboxu, pomocí které lze identifikovat přítomnost viru.[25]

Dále se do používaných metod antivirovými programy zahrnuje **whitelisting**. Tato metoda však neslouží k identifikaci virů, nýbrž k zabránění spuštění nežádoucích programů. Metoda využívá tzv. whitelist, který obsahuje programy, které byly předem prověřeny uživatelem počítače nebo výrobcem antiviru. Při zapnutí téhle služby dochází při spuštění programu ke kontrole, zdali se daný program nachází ve whitelistu a až poté je buď spuštěn, nebo je mu spuštění zamítnuto. Jedná se o velmi bezpečnou metodu, která však může být v jistých situacích nevhodná, protože zabraňuje spouštět cokoliv, co není obsaženo v seznamu výjimek.

Všechny z uvedených metod by například vůbec nevedli k odhalení rootkitů, protože pracují jiným způsobem. Pro jejich detekci je zapotřebí monitorovat chování zařízení, protože rootkity většinou zasahují do hlubší části systému. Podrobnější popis, jakým antiviry postupují při hledání rootkitu, bude následovat v nadcházející kapitole u jednoho z uvedených programů. Odstranění rootkitu je obtížné a mnohdy to nevede k jinému řešení než k přeinstalaci celého operačního systému.

Kapitola 4

Antivirové programy současnosti

To, jaké způsoby využívají antivirové programy k identifikaci, již bylo rozebráno a nyní bude následovat výčet konkrétních antivirů od různých firem. Výčet antivirů byl zaměřen především na ty, které podporují operační systém Linux a mohli by se tak hodit pro zhotovení této práce. U každého z uvedených antivirů bude podrobnější popis zaměřen na to, který z uvedených metod využívají k identifikaci virů, jaké výsledky dosahují při testování a jak si tedy obstávají mezi ostatními antiviry.

4.1 ClamAV

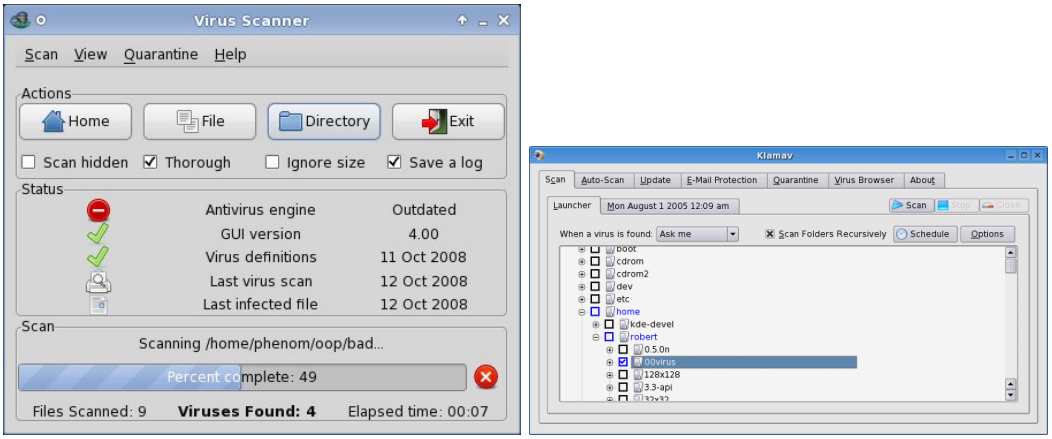
Jedním z nejznámějších a nejpoužívanějších antivirových balíčků pro Linux je právě ClamAV. Jedná se o antivir, který je dostupný zcela zdarma ke stažení buďto na oficiálních stránkách vývojáře nebo přes repositář systémových balíčků. Původně byl balíček vyvinut pouze pro platformy založené na Unixovém jádře. V dnešní době je však balíček dostupný i pro systém Microsoft Windows.

Antivirus slouží k identifikaci virů, trojských koňů, ransomwaru, červů a dalšího malwaru. Nejčastěji se však využívá pro identifikaci elektronické pošty z poštovních klientů, která obsahuje v příloze infikovaný soubor. ClamAV dokáže pracovat s různými druhy spustitelných souborů, dokumentů, formátů uložišť pro elektronickou poštu a dokáže také kontrolovat archivy, a to i rekurzivně.

Pro aplikaci existují dvě uživatelská rozhraní pro Linux. Nazývají se KlamAV a ClamTk. KlamAV je, co se obsahem funkcí týče, daleko propracovanější a je možné přes něj využít plnohodnotně antivirový program ClamAV. Ukázka uživatelského rozhraní je vyobrazena na obrázku 4.1a. Na druhou stranu ClamTk, které je na obrázku 4.1b, klade důraz na jednoduchost a je omezen pouze na spuštění ručních testů a správu karantény. Obě uživatelské rozhraní se však podotýkají s velmi nízkou kvalitou českého překladu, a tak je vhodnější je používat v anglickém jazyce.

Balíček celkově obsahuje tři různé nástroje. Jedním z nejpoužívanějších nástrojů z nabídky je skener, který je ovladatelný pomocí příkazové řádky. Jako většina antivirových skenerů lze nástroj spustit pouze na vyžádání uživatele. Tím je zajištěno, že nástroj nespotřebovává čas procesoru bez vědomí uživatele.

Pro identifikaci škodlivého softwaru nástroj využívá první z uvedených metod. Pro připomenutí se jedná o metodu, která ke svému běhu využívá virovou databázi na porovnávání skenovaných souborů. Aktualizaci téhle databáze má na starosti další ze tří nástrojů balíčku, který se nazývá Freshclam. Pro dosažení nejlepších výsledků skenování, je zapotřebí



(a) ClamTk (b) KlamAV

Obrázek 4.1: Ukázka uživatelských rozhraní

zajistit co nejaktuálnější databázi virů. Podle sdílených statistik se databáze aktualizuje nejméně každé 4 hodiny a obsahuje již přes 5 760 000 existujících virů. Statisticky každý den v databázi přibude průměrně 23 040 nových odhalených virů.

Jako poslední ze tří nástrojů se nabízí více-jádrový skenovací démon, který dokáže běžet na pozadí bez potřebného zásahu uživatele a nepřetržitě tak hledat skrytou hrozbu. Zde však již dochází k vytěžování procesoru. Pro urychlení skenování lze programy s démonem propojit. Nástroj v systému vytvoří nového uživatele *clamav*, který má povoleno skenovat i systémové složky. ClamAV má také podporu Milter rozhraní pro program Sendmail, který zajišťuje přepravu elektronické pošty. Pomocí toho má aplikace možnost filtrovat příchozí elektronickou poštu.[18][29]

4.1.1 Efektivnost

V dnešní době dochází k vyhodnocování různých antivirových testů téměř denně. Testy jsou nejčastěji prováděny skupinou Shadowserver, která se zaměřuje na zlepšování kvality bezpečnosti na internetu. V roce 2010 bylo na ClamAV testováno 22 milionů virových kódů a byla zaznamenána úspěšnost 76.64%. S tímto výsledkem se program zařadil na 9. místo z celkově 19 testovaných antivirových programů. Přestože se program nezařadil mezi první příčky, stále jde o lepší výsledky než jakého dosáhly déle zavedené antivirové programy.

V roce 2011 se ClamAV podrobil šestiměsíčnímu testování, které bylo opět provedeno skupinou Shadowserver, a kde dosáhl detekci 75.45% a tím se zařadil na 4. místě za programy AhnLab, Avira, BitDefender a Avast. Pro zajímavost lze uvést, že AhnLab dosáhl úspěšnosti 80.28% a tím obsadil 1. místo. Ačkoliv činí rozdíl pouze 5%, v kontextu testování milionu vzorků jde o vysoké číslo.[46][47]

4.1.2 Výhody a nevýhody

Mezi hlavní výhodu programu patří, že je program zcela dostupný bezplatně a má širokou podporu v rámci platforem. Jedna z dalších výhod je také v aktuálnosti virové databáze a více-jádrovém skenovacím démonu.[19]

Nevýhodou ClamAV je jeho rychlost skenování. Oproti běžným antivirovým programům je až desetkrát pomalejší. Pokud již dochází ke skenování souborů, doporučuje se skenovat

co největší část disku najednou a tím eliminovat načítání virové databáze na minimum, které je zdlouhavé. Díky pomalosti programu pak dochází k nízkému využití kontroly on-access, která skenuje soubor při jeho otevření. Takové skenování by trvalo nepřiměřeně dlouho. Jedna z nevýhod aplikace může být i to, že nenabízí vyléčení poškozených neboli infikovaných souborů. Aplikace rovnou soubor odstraní, anebo jej přesune do karantény. Takové řešení někdy může vést ke ztrátě dat.

4.2 Chkrootkit

Jedná se o program, který se liší od běžných antivirových programů tím, že neidentifikuje viry, trojské koně nebo červi, ale zaměřuje se na rootkity. Pro svou funkcionalitu využívá běžně dostupných unixových nástrojů jako jsou například *strings* nebo *grep*. Díky tomu je program omezen pouze na unixové systémy. Je zcela bezplatně dostupný. Celkově se jedná o shell script, který za využití unixových nástrojů prohledává jádrové systémové programy a jejich podpisy. Následně dochází k porovnání traverzu souborového systému */proc* s výstupem příkazu *ps* (process status, který zobrazuje aktuálně běžící procesy a informace o nich) pro hledání nesrovnalostí. Za pomocí toho může být odhalen běžící rootkit, který se většinou skrývá pod názvy jiných systémových procesů.

Nástroj je možné spustit z *data recovery* módu, například pomocí *LiveCD*, a nebo ho lze spustit z alternativní složky, ze které může spustit všechny své vlastní příkazy. Použitím alternativní složky je možné zvýšit úspěšnost identifikace rootkitu, protože program použije vlastní nástroje, nikoliv ty, které jsou integrovány v systému. Důvod je takový, že i tyhle systémové nástroje můžou být pod vlivem rootkitu a ovlivnit tak výsledek.

Na rozdíl od předešlého nástroje ClamAV, Chkrootkit neobsahuje žádné grafické uživatelské rozhraní a je tak pouze plně ovladatelný pomocí příkazové řádky. Tím se může jevit pro určitou skupinu lidí těžce ovladatelný. Umožňuje spustit hromadnou kontrolu souborů, které jsou popsány níže, anebo pomocí přepínačů lze zvolit pouze kontrolu, která je uživatelem vyžadována.

Jak již bylo zmíněno výše, jedná se o pouhý shell script, který je přítomný pro jednodušší ovladatelnost. Po spuštění skriptu dojde postupně k prohledání binárních souborů a hledání rootkitu. Skript se celkově skládá z 8 zdrojových kódů, kde se každý zaměřuje na jinou část skeonvání. Jednotlivé zdrojové kódy a jejich zastoupení při vyhledávání jsou rozebrány níže.[57][37]

- **ifpromisc.c**: zkontroluje, zda-li je síťová karta nastavena do promiskuitního režimu
- **chklastlog.c**: zkontroluje mazání lastlogu, ve kterém jsou uloženy přihlášení uživatelů počítače
- **chkwtmp.c**: zkontroluje mazání wtmp souboru, který taktéž loguje přihlášení uživatelů počítače
- **chkutmp.c**: zkontroluje mazání utmp souboru
- **check_wtmpx.c**: zkontroluje mazání wmtpx souboru (pouze pro Solaris)
- **chkproc.c**: zkontroluje příznaky LKM trojanů
- **chkdirs.c**: zkontroluje příznaky LKM trojanů
- **strings.c**: zkontroluje rychlé a nežádoucí nahrazení textových řetězců

[37]

4.2.1 Výhody a nevýhody

Pro zkušené uživatele se ovladatelnost příkazovou řádkou může zdát jako výhoda, pro nezkušené jako nevýhoda. Pro vypracování této práce je to například nutná podmínka. Další výhodou je pak podpora skenování připojených složek pomocí protokolu NFS.

Mezi nevýhody je možnost zařadit to, že program pouze detekuje možnou hrozbu ze strany rootkitů, ale dále již neřeší jeho následné odstranění ze zařízení. Občas se může stát, že nástroj nahlásí infikovaný soubor, ale nemusí se jednat přímo o rootkit. Může se jednat o běžný systémový stav. V takovém případě je vhodné, pokud dojde k porovnání aktuálního výstupu programu s výstupem, který byl získán při spuštění programu na čisté instalaci systému. To umožní zjistit, zdali se jedná o běžný stav systému, anebo došlo v průběhu práce k nějaké změně.

4.3 Sophos

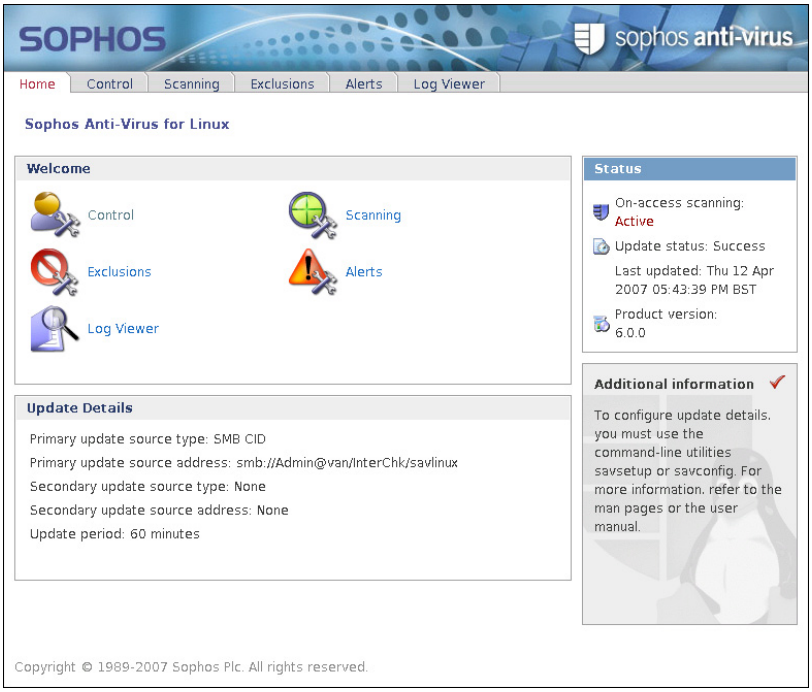
Jedním z dalších antivirových programů, který byl vybrán, je od firmy Sophos. Firma na trh antivirů nabízí svůj program Sophos Endpoint Protection. Program je dostupný pro platformy Windows, MacOS i Linux. Nejedná se však o freeware software. Pro Linux je dostupná verze, která je volně ke stažení, avšak pro ostatní platformy je program placený.

Hlavní ochrana se opět zaměřuje na viry, trojany a pokročilé hrozby jako jsou cílené útoky. Při svém běhu kontroluje podvodné internetové stránky, pravost certifikátů u aplikací, zařízení a data. Antivirus nabízí flexibilní nasazení, kde je k výběru cloudová nebo lokální správa programu. Program obsahuje i kvalitní uživatelské prostředí. Jak již bylo zmíněno, Sophos Endpoint Protection provádí webovou kontrolu, kdy prohledává všechna příchozí HTML data a případně blokuje přístup k podvodným stránkám všeho druhu.[17] V rámci dostupných výsledků testů se při blokování stránek antivirus prokázal velmi kvalitním výsledkem. Při testu se mu podařilo odhalit 90% testovaných škodlivých stránek. Co se týče detekce jiného škodlivého softwaru, Sophos nedosahuje v téhle oblasti příliš kvalitních výsledků. Program zahájí skenování až při otvírání složky/souboru, kde se škodlivý software nachází. V nabídce je na výběr také on-demand skenování. To nemění nic na tom, že při skenování programem došlo k odhalení pouze 61% přítomného malwaru. Poté bylo zbylých 39% nakažených souborů otevřeno či spuštěno. V téhle fázi dokázal Sophos detekovat 13% ze spuštěných souborů. Zbylých 26% dokázalo bez detekování proniknout do systému.[45]

4.3.1 Výhody a nevýhody

Velmi překvapivý byl výsledek dosažený v detekování podvodných a nedůvěryhodných stránek. Mezi výhody lze zařadit i jednoduché a snadno ovladatelné uživatelské rozhraní. Náhled na uživatelské rozhraní se nachází na obrázku 4.2. Jako poslední výhodu lze uvést kvalitně zpracované dálkové monitorování a správa programu.

Obrovskou nevýhodou se jeví velmi slabé výsledky při detekci malwaru. Za nevýhodu, která nijak neovlivňuje výsledky antiviru, může být považováno neustále vyžadování potvrzení UAC v systému Windows při spuštění uživatelského rozhraní. Jedná se o povýšení aplikace do administrátorského režimu. UAC požadavek je možné v systému Windows jistým způsobem omezit, ale nedoporučuje se to běžným uživatelům, protože omezení se vztahují na celou skupinu aplikací, nikoliv pouze na jednu konkrétní.[51]



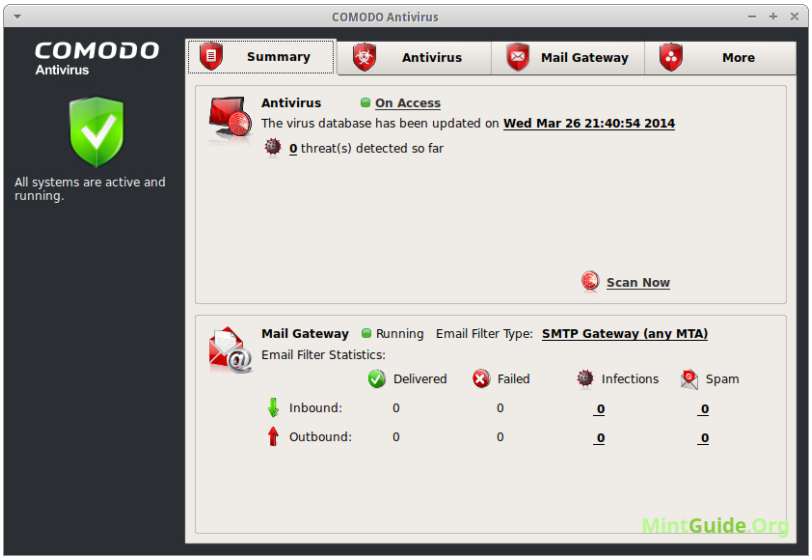
Obrázek 4.2: Ukázka uživatelského rozhraní pro Sophos

4.4 Comodo

Mezi další známé antivirové programy patří Comodo Antivirus. Jedná se o antivirový program, který je dostupný jak pro unixové prostřední, tak pro Microsoft Windows, MacOS, nebo dokonce i pro telefony, které běží na operačním systému Android. V základní verzi je zcela bezplatný a nabízí poměrně rozsáhlou ochranu. Ukázku uživatelského rozhraní pro antivirus Comodo lze najít na obrázku 4.3.

Comodo Antivirus chrání uživatele před různým malwarem, jako jsou viry, spyware nebo rootkity. Umožňuje kontrolu síťových a pevných disků. Kontroluje také i externí uložiště připojené pomocí konektoru USB. Antivirus má však pod kontrolou i tok sítě. Tuhle kontrolu zajišťuje osobní firewall, ve kterém lze nastavovat jak příchozí, tak i odchozí omezující pravidla. Další bezpečnostní mechanismus obsažený v programu je Sandbox, který zajišťuje, jak již bylo řečeno, oddělování běžících procesů, čímž řídí přístup k diskům a síti. Poslední ochranou, kterou Comodo nabízí, je systém pro odhalení průniku, který monitoruje síťový provoz a snaží se odhalit podezřelé aktivity. Pokud dojde ke spuštění aplikace, program zkontroluje její ověření a před samotným spuštěním upozorní na neznámou aplikaci. K tomuto rozlišování, známých a neznámých, aplikací využívá metodu *whitelisting*, neboli seznam bezpečných souborů a aplikací, který je pravidelně doplňován výrobcem, anebo je zapotřebí ruční přidání aplikace uživatelem. Všechny soubory a aplikace obsažené v listu jsou automaticky spuštěny bez předchozího upozornění. Jednou za zajímavostí, kterou program poskytuje, je možnost drag and drop pro skenování souborů.

Za nadstandardní výbavu, kterou program nabízí, lze označit možnost vytvoření záchranného disku. Pokud se systém objeví v situaci, kdy je napaden malwarem a nedaří se ho odstranit, lze načíst systém ze záchranného disku. Disk je však zapotřebí vytvořit dříve, než se systém dostane do takové situace. Jedná se o takový bod obnovení nebo záložnou bitovou kopii.[23]



Obrázek 4.3: Ukázka uživatelského rozhraní pro Comodo

4.4.1 Výhody a nevýhody

Antivirus dosahuje vysoce kvalitních výsledků při blokování malwaru. Při spuštění neznámých programů, které nejsou obsaženy ve *white-listu*, využívá služby Sandbox. Firma Comodo se i velmi kvalitně stará o virovou databázi a snaží se ji udržet nejaktuálnější.

Comodo Antivirus neposkytuje žádnou ochranu proti podvodným internetovým stránkám, protože firma Comodo využívá samostatný program pro webovou ochranu. Pokud je uživatelem vyžadována i tahle ochrana, je zapotřebí mít na zařízení nainstalován další program.[43]

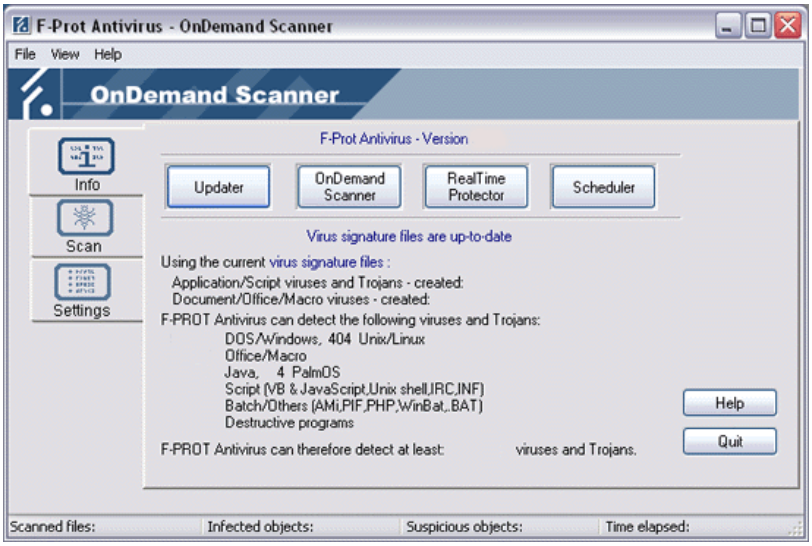
4.5 F-Prot

Antivirový program F-Prot postupně upadá do zapomnění. Od roku 2011 od vývojářů nepřišla žádná nová aktualizace programu. Virová databáze je ale stále aktualizována, a tak je antivir stále funkční pro unixové systémy i pro systémy Microsoft Windows. I po tak dlouhé době bez aktualizací není program dostupný zdarma ke stažení. Na stránkách výrobce lze stáhnout zkušební třicetidenní verzi. Ukázka uživatelského rozhraní je vyobrazena na obrázku 4.4.

Dostupná základní verze antiviru F-Prot dodává poměrně spolehlivé, jednoduché na ovládání a účinné zabezpečení vašeho počítače. Program nikdy nebyl testován předními společnosti AV-Test nebo Shadowserver, takže ho nelze pořádně porovnat s ostatními antiviry. Každá vydaná verze do roku 2011 však dosáhla 100% výsledků v detekování malwaru při testu skupiny Virus Bulletin, která sídlí pod firmou Sophos a vede svůj vlastní magazín o testování antivirových programů.

Program zajišťuje ochranu počítače obdobnou jako ostatní antiviry. Obrovskou výhodou tohoto antiviru je, že využívá druhou z předchozích uvedených metod a tím je detekce pomocí heuristiky. Tím umožňuje detekovat nové viry, které nemá obsažené v databázi. Pravděpodobně díky této heuristické detekci dokáže zajistit kvalitních výsledků při testech. Další neobvyklá vlastnost je heslem zabezpečené nastavení antivirového programu, aby došlo k zabránění k možným útokům na samotný antivirový program. Jako většina antivirů

obsahuje integrovanou ochranu v rámci elektronické pošty, která kontroluje jak příchozí, tak odchozí poštu, ale také přílohy elektronické pošty.[12]



Obrázek 4.4: Ukázka uživatelského rozhraní pro F-Prot

4.5.1 Výhody a nevýhody

Velké plus u tohoto antivirového programu mají především velmi dobré výsledky z testů. Výsledky má na svědomí jak kvalitní aktuálnost virové databáze, tak právě zmiňovaná heuristická detekce, která je méně obvyklá u běžných antivirových programů.

Nevýhodou jsou aktualizace programu, které zamrzly v roce 2011, ale přesto je program stále placený. Díky neaktuálnosti je problém s ochranou elektronické pošty, která je převážně pro ty, co využívají k přístupu na internet prohlížeč Internet Explorer.

4.6 Avast

Avast Antivirus se na trhu objevuje již od roku 1988. Program je vyvíjen českou společností Avast Software s.r.o. Přestože je program vyvíjen českou společností, má poměrně rozsáhlé působení a obsahuje 45 dostupných jazykových verzí. Co se týče platformové podpory, antivirus je k dostání pro systém Microsoft Windows, ale také zařízení od firmy Apple a Android. Na oficiálních webových stránkách lze stáhnout i verzi pro unixové systémy, která je nicméně již neaktuální. Avast má ve své nabídce jak placenou verzi, tak freeware verzi programu.

Zajišťuje ochranu před různými druhy malwaru. Většina freeware dostupných antivirů v dnešní době obsahuje pouze základní ochranu. Avast Free Antivirus ale oproti konkurenci nabízí daleko více. Ve své freeware verzi obsahuje standardní ochranný štít, antivirové a antispywarové vlákno, rezidentní ochranu před rootkity, http(s) ochranu, základní ochranu proti známým hrozbám na síti, boot-time skenování, aktuální virovou databázi, herní režim, blokování podezřelých a podvodných (phising) stránek, kontrolu elektronické pošty atp.[35]

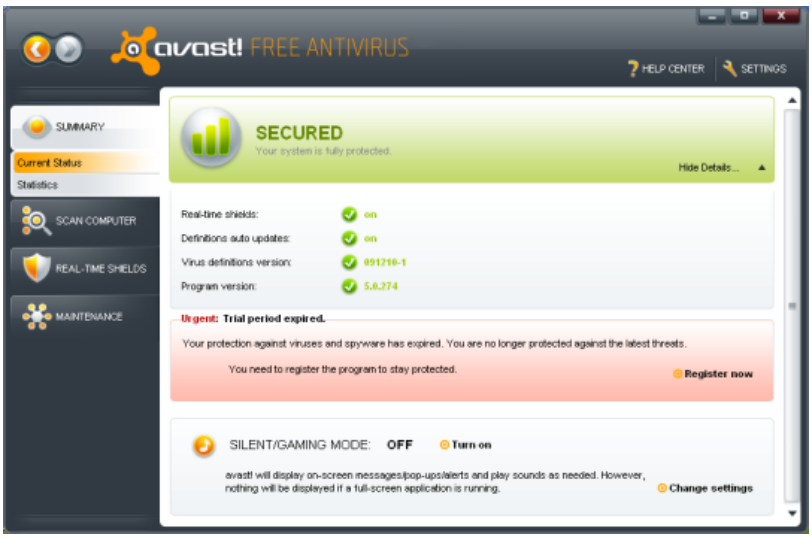
Co se týče hodnocení programu v testech, dosahuje Avast Free Antivirus kvalitních výsledků. V testech od společnosti AV-Test získal 15.5 bodů z celkových 18 možných. V rámci

testu, od společnosti AV-Test, se testuje ochrana, výkon (jak dlouho trvá programu skenování a identifikace) a použitelnost. Avast Free Antivirus získal plný počet v testování ochrany a použitelnosti. Nicméně pokulhává v rychlosti, a tak v kategorii výkonu došlo ke ztrátě 2.5 bodu.
Svoji kvalitu si potvrdil i v dalším testu, kdy uspěl v 97% případů. Při tomhle testování byly použity různé viry, trojské koně, ransomware a potenciální nechtěné aplikace (Zkráceně se jim říká PUBs. Pro skenování PUBs je potřeba změnit nastavení programu, protože implicitně je tohle skenování vypnuto). Při testech na blokování nežádoucích a podvodných stránek a stažení škodlivých souborů, dokázal Avast zablokovat 87% případů. S tímhle výsledkem se tak řadí ihned za nejefektivnější antivirové programy jako jsou Norton, Bitdefender nebo Kaspersky.[42]

K identifikaci malwaru nepoužívá real-time skenování, ale skenování zahájí až těsně před spuštěním programu. Využívá klasického principu porovnávání souborů s databází již známých škodlivých softwarů. Jako většina antivirových programů obsahuje on-demand skenování, kde je možnost zvolit, které diskové oddíly se mají skenovat. Případně je možnost skenování omezit pouze na určité složky či soubory.

Pokud k zařízení připojíme externí periferii, tak Avast neprovádí automatickou kontrolu.

Jednu z neposledních zajímavostí, které antivirus obsahuje, je Wi – Fi inspektor, který dokáže proskenovat všechna zařízení na síti a zkontrolovat tak jejich nastavení. Po skenování Avast zobrazí výsledek, kde dojde k upozornění na špatná nastavení. Nad zařízeními ale nemá takovou kontrolu, aby špatná nastavení dokázal opravit, proto je to již ponecháno na uživateli.[35][55]



Obrázek 4.5: Ukázka uživatelského rozhraní pro Avast

4.6.1 Výhody a nevýhody

Mezi hlavní program firmy Avast, který přitahuje nové uživatele, je freeware verze Avast Free Antivirus, která je k dostání ke stažení na oficiálních stránkách firmy Avast. Poskytuje širokou škálu ochran, které dosahují kvalitních výsledků. Jako jeden z mála freeware antivirových programů obsahuje správce hesel, který dokáže chránit hesla ve webovém prohlížeči.

Nevýhodou je, že podle testů dosahuje slabých výsledků v oblasti výkonu, a tak skenování zabírá delší dobu. Z pohledu běžného uživatele se může jednat o zanedbatelnou dobu. Za nevýhodu lze také označit, že antivir neprovádí automatickou kontrolu, pokud je k zařízení připojena externí periferie.

4.7 ESET NOD32 Antivirus

Předešlý uvedený antivirový program byl domácí výroby, nynější je naopak výroby slovenské. Firma ESET působí na trhu již od roku 1992 a zabývá se bezpečností v oblasti informačních technologií. Nabízí větší množství programů, jak již pro domácnosti, tak pro firmy. Její produkty jsou dostupné pro systém Windows, MacOS, Linux a dokonce i pro mobilní platformu Android. Nejedná se však o freeware verze, a tak všechny produkty vyžadují zakoupení licence.

ESET NOD32 Antivirus poskytuje ochranu proti všem typům online i offline hrozeb a brání šíření škodlivého kódu. Brání počítač před speciálním druhem malwaru, který je nazýván ransomware. Nově také zajišťuje ochranu před útoky pomocí skriptů, které se snaží zneužít Windows PowerShell. Tahle ochrana zároveň zahrnuje detekci škodlivých JavaScriptových útoků v prohlížečích Mozilla Firefox, Google Chrome, Microsoft Internet Explorer a Microsoft Edge. V rámci ochrany, ze strany internetu, zahrnuje Anti-Phising, který zabraňuje podvodným internetovým stránkám získat citlivá data počítače. Dále kontroluje soubory již v průběhu stahování, čímž snižuje čas kontroly. Tuhle kontrolu provádí pouze u určitých typů souborů, například u komprimovaných archivů. Další novinkou je pak ochrana před útoky již při startování systému Windows. Ochrana však platí pouze na systémech s rozhraním UEFI.

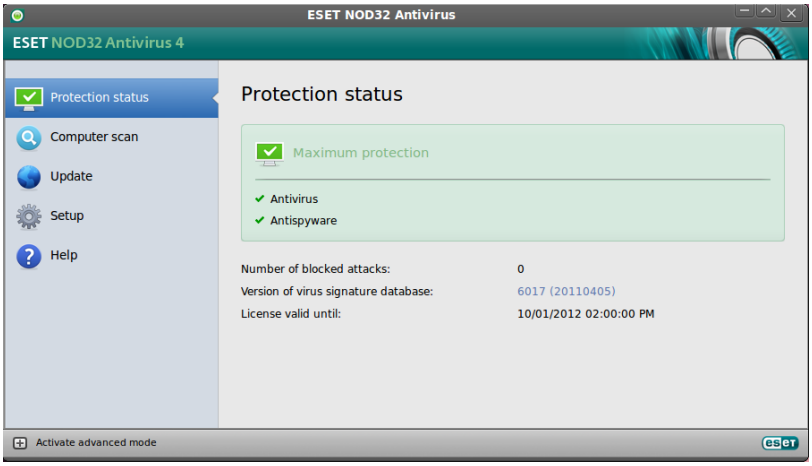
Kromě nespočetného množství bezpečnostních ochran ESET NOD32 Antivirus také vyniká s nízkými systémovými nároky. Veškerá automatická kontrola, která probíhá bez vědomí uživatele, je velmi kvalitně naplánována tak, aby nevytěžovala procesor při práci uživatele. To je také zajištěno tím, že program je napsán z velké většiny v jazyce symbolických adres. Antivirus také obsahuje herní režim, který odkládá všechna upozornění o aktualizacích, systémových nebo jiných aktivitách, pokud je spuštěna celoobrazovková aplikace.[22]

Celkově se na identifikaci škodlivého softwaru podílí on-demand skener a čtyři různé real-time monitory. On-demand skener obsahuje plánovač, který ho dokáže zapnout ve stanovenou hodinu. Jinak je skenování pouze v rukou uživatele. Každý z real-time monitorů má za úkol identifikaci jiného druhu škodlivého softwaru podle rozdělení níže:

- **AMON** (Antivirus Monitor) - provádí ochranu systému před viry
- **DMON** (Document Monitor) - zaměřuje se na skenování dokumentů aplikace Microsoft Office a dalších souborů proti macrovirům
- **IMON** (Internet Monitor) - před jakýmkoliv uložením souboru staženého z internetu provede kontrolu proti virům, zabrání tak uložení viru na disk
- **EMON** (E-mail Monitor) - skenuje příchozí a odchozí elektronickou poštu přes MAPI rozhraní
- **XMON** (MS Exchange Monitor) - skenuje odchozí a příchozí poštu pro Microsoft Exchange Server, funguje pouze v serverovém prostředí a tento monitor není součástí balíčku pro běžné pracovní stanice

[56]

Při testech společnosti AV-Test, kde opět došlo k hodnocení tří kategorií, které jsou ochrana, výkon a použitelnost, ve stejném pořadí ESET NOD32 Antivirus získal 5.0, 4.5 a 6.0 bodů. Což vedlo k celkovému součtu 15.5 bodů z 18 možných. Před NOD32 se zařadily pouze antivirové programy Kaspersky, Bitdefender Antivirus Plus a Trend Micro.[44]



Obrázek 4.6: Ukázka uživatelského rozhraní pro ESET NOD32

4.7.1 Výhody a nevýhody

Jedná se o poměrně kvalitně zpracovaný antivirový program, který dosahuje více než dobrých výsledků v rámci testů. Antivirus obsahuje velmi dobře zpracovanou ochranu při startování systému Windows. Skenování disků probíhá poměrně v krátké časové době, záleží tedy na velikost dat na discích. Kvalitně zpracované je i uživatelské rozhraní s velkou možností různých nastavení, které se může pro některé uživatele jevit až příliš podrobné a tím i složité.[44]

Kapitola 5

Raspberry Pi

Tahle kapitola bude zcela nesouvisející s viry a antivirovými programy a bude se zaměřovat na jednodeskový počítač, na kterém poběží výsledný implementovaný detektor. Jedná se právě o Raspberry Pi, které vzniklo roku 2012 pod britskou nadací Raspberry Pi Foundation. Prvotním cílem jejich projektu bylo vytvořit finančně dostupné zařízení, které by pomohlo ve školách studentům pochopit základy hardwaru a rozvinout programovací myšlení. Díky své nízké ceně na trhu a jeho velikosti, která je přirovnatelná ke dnešním platebním kartám, si ale Raspberry Pi neoblíbily pouze školní vzdělávací programy, ale také nadšenci pro různorodé elektronické projekty, které vyžadují více než obyčejný mikrokontroler. Od roku 2012 již bylo vyvinuto několik verzí tohoto jednodeskového počítače, které budou popsány níže.

Raspberry Pi je ve skutečnosti jednočipový počítač. Jednočipový počítač se skládá z integrovaný obvods, který ve svém pouzdře většinou obsahuje vše potřebné, aby dokázal vykonávat určitou aplikaci bez použití dalších podpůrných obvodů. Pouzdro integrovaný obvods ve většině případů obsahuje paměť pro uložení programu, která bývá typu FLASH, EEPROM nebo ROM, a paměť typu RAM pro ukládání výpočtů a mezi výpočtů aplikace. Nedílnou součástí jednočipových počítačů jsou v dnešní době také bloky pro analogové vstupy a výstupy a další aplikační logika.

Tím bylo nastíněno, z čeho je takové Raspberry Pi vlastně vytvořené. Jedná se tedy o plnohodnotný stolní počítač, který sice nedosahuje takových výkonů, ale je možné k němu připojit potřebné periferie jako je monitor, myš a klávesnice. Počítač většinou obsahuje operační systém a výjimkou tomu není ani u Raspberry Pi, na který lze nainstalovat různé distribuce Linuxu, RISC OS nebo Microsoft Windows 10 IoT Core. Nejčastěji se používají Linuxové operační systémy a jedním z nejznámější je Raspbian, který je založen na operačním systému Debian. Jedná se také o oficiální systém podporovaný společností Raspberry Pi Foundation. Dalším nejpoužívanějším je Pidora, který je založen na operačním systému Fedora. Dostupných systémů pro tuhle platformu je daleko více a je jen na uživateli, který si vybere. Je velice výhodné vybírat operační systém podle zkušeností s daným systémem a také podle dostupnosti potřebných nástrojů pro vytvářenou aplikaci.

5.1 Verze

Poslední verze Raspberry Pi byla vydána v roce 2016. I když je nejnovější verze daleko výkonnější než desky předešlé, staré verze jsou stále dostupné, protože je o ně na trhu neustále zájem. To je z toho důvodu, že ne vždy je zapotřebí pro běh aplikace výkon, a

proto je vhodné občas koupit starší verzi, která ušetří peníze, spotřebu a třeba i místo. Nyní následují rozbory jednotlivých verzí, kde bude uvedeno, jaký hardware nabízí.

5.1.1 Raspberry Pi

Jedná se o úplně první verzi, která obsahuje procesor BCM2835 ARM Cortex-A6. Procesor běží na 700MHz. Jako grafický čip slouží VideoCore IV, který podporuje OpenGL ES 2.0, 1080p 30fps a MPEG-4. K připojení monitoru, jako grafického výstupu, se nabízí rozhraní Composite RCA, HDMI nebo DSI, čímž je zajištěno bezproblémové připojení většiny displejů nebo monitorů. Pokud je zapotřebí pomocí Raspberry přehrávat hudbu, nabízí se pro připojení reproduktorů klasický 3,5mm konektor, anebo lze využít i přenosu zvuku přes HDMI. Pro komunikaci s vnějšími zařízeními je dostupných 12 portů, které lze nastavit buď jako vstupy nebo výstupy. Pro sériovou komunikaci je možné využít UART sběrnici, I2C anebo SPI. Později byla přidána periferie Watchdog Timer, která hlídá systém. Pokud se systém zacyklí, periferie zajistí jeho restartování.

První verze byla vydána ještě v různých modelech. Nejdříve byl dostupný **Model B**, který vyšel v roce 2012. Model B obsahoval původně 256 MB operační paměti RAM. Později byla operační paměť zvednuta na 512 MB. Paměť RAM je sdílená s grafickým čipem. Pro připojení externího uložiště je možné použít SD nebo MMC kartu. Obsahuje celkově 2 USB porty, přes které je možnost připojit periferie. Pokud je zapotřebí využití internetu, je dostupný ethernetový adaptér s běžným konektorem RJ45.
Po Modelu B přišel o rok později na trh **Model A**, který se liší pouze ve velikosti paměti RAM, která byla vrácena na původních 256 MB, byl oddělán jeden USB port a ethernetové rozhraní pro připojení do sítě. O rok později přišly na trh další dva modely. **Model A+**, který je podobný Modelu A, ale jako externí uložiště lze nyní využít micro SD kartu. **Model B+** sdílí s grafickým čipem 512 MB paměti RAM, stejně jako druhý „pluskový“ model obsahuje slot na micro SD kartu, ethernetový kabel a celkově čtyři USB porty.[39]

5.1.2 Raspberry Pi 2

V roce 2015 byla vydána verze 2. Tahle verze byla dosud vydána pouze v modelu B. Deska je opět poháněna procesorem od Broadcom, přesněji BCM2836 ARM Cortex-A7. Jedná se o čtyřjádrový procesor nataktovaný na 900 MHz. Uvádí se, že tahle verze je až 6x výkonnější než předešlá verze nejvyššího modelu B+. Operační paměť byla navýšena na 1 GB. Co se týče grafického čipu, ten zůstal nedotčený, a tak stále obsahuje VideoCore IV se stejnou podporou. Ostatní výbava je podobná modelu B+ první verze. Byly zachovány čtyři USB porty, ethernetový adaptér s konektorem RJ45, stejný počet nastavitelných vstupních/výstupních portů a slot pro micro SD kartu. Pokud se jedná o zpětnou kompatibilitu, měla by být bezproblémová s prvotní verzí desky.[39]

5.1.3 Raspberry Pi 3

Stejně jako předchozí verze je dostupná pouze jako Model B. Jako první ze všech verzí byla deska osazena 64 bitovým procesorem nataktovaným na 1,2 GHz. To je o 33% vyšší frekvence procesoru. Zároveň se jedná o modernější jádro s efektivnější instrukční sadou. Řeč je přímo o čtyřjádrovém procesoru ARM Cortex-A53. Uvádí se, že při běžném používání by měla být verze 3 o 50% rychlejší než její předchůdce. Druhou největší přidanou novinkou je modul WiFi pro bezdrátové připojení k síti. Pro bezdrátovou komunikaci byl také přidán modul podporující standard Bluetooth. Velikosti, rozložení a počet ostatní výbavy desky

byly zachovány dle původní verze. Malou změnou konstrukce si prošel pouze slot na micro SD kartu.[39] Verze Raspberry Pi 3 bude také využita k implementaci práce.

Obrázek 5.1: Raspberry Pi 3

5.1.4 Raspberry Pi Zero

Jedná se zatím o poslední vydanou verzi. Dostupná je ve dvou modelech, a to Zero a Zero W. Oproti předešlým verzím se jedná o levnější a menší variantu. Jeho velikost je přibližně o půlku menší než deska předešlých verzí. Deska je poháněna procesorem BCM2835 nataktovaným na 1GHz. Jako operační paměť musí vystačit 512 MB RAM a grafický čip zůstává stejný jako ve všech verzích. I přes svou velikost stále disponuje se 40 piny pro vnější komunikaci. Piny však nejsou osazeny. Kvůli velikosti byly odstraněny porty DSI pro grafický výstup, ethernet pro připojení internetu, analogový audio výstup a klasické USB porty. Pokud je zapotřebí k desce připojit grafický výstup, je k dispozici pouze mini HDMI port, který by měl být dostačující. Pro napájení desky je třeba použít micro USB port. Pomocí micro USB portu je řešena i komunikace s počítačem při programování. Rozdíl v uvedených dvou modelech se skrývá pouze v tom, že model Zero W navíc obsahuje vestavěnou Wi-Fi a Bluetooth pro snazší bezdrátovou komunikaci.[39]

Kapitola 6

Implementace

Všechny doposud uvedené informace lze využít k implementaci samotné aplikace. Jedná se tedy o automatický detektor virů pro USB uložiště. Nyní budete tedy blíže rozebrána praktičtější část práce. Nejdříve budou popsána použitá zařízení a poté množina programovacích jazyků, která lze využít k implementaci. Z uvedených jazyků bude podrobněji rozebrán pouze jeden, který bude následně použit k programování. Dále budou uvedeny potřebné knihovny a frameworky (česky se můžou vyskytovat i pod názvem rámce, ale v IT světě se většina názvů nepřekládá), které jsou nezbytné pro zhotovení práce.

6.1 Vývojové prostředí

Aplikace by bylo možné vyvíjet i pro běžné stolní počítače. Cílem práce je však vytvoření přenosného detektorů virů, který je zaměřen na to, aby došlo k zajištění bezpečného připojení USB uložiště k osobnímu počítači. Aby bylo těchto požadavků docíleno, je zapotřebí skenování provést mimo osobní počítač. K tomu bude využit právě jednodeskový počítač a v tomto případě se bude jednat právě o zmiňované Raspberry Pi. Při výběru verze nejde o spotřebu elektrické energie, ale spíše o výkon kvůli rychlosti skenování, a proto je vhodnější zvolit nejnovější verzi Raspberry Pi 3. Využít by se samozřejmě dali i jiné jednodeskové počítače, které by běžely na stejné architektuře a tou je ARM. Postupně však budou zjištěny různá úskalí tohoto jednodeskového počítače. Z těchto důvodů zde lze uvést i možný alternativní výběr. Jelikož ARM architektura omezuje dostupné zdroje, lze využít například jednodeskové počítače SolidPC Q4[48], AMD Gizmo Board anebo Intel Galileo. Existuje jich však větší množství. Jedná se o jednodeskové počítače, které jsou založeny na jiné než ARM architektuře. Problém bude podrobněji rozebrán v jedné ze sekcí níže.

Aby bylo k aplikace možné přidat i nějaké rozšiřující funkce, je vhodné, když k ní bude vytvořeno grafické uživatelské rozhraní. Pokud by bylo grafické rozhraní vynecháno, lze využít například zvukové signalizace na infikované/neinfikované USB uložiště. Cílem práce je však i grafické rozhraní, aby bylo možné zobrazit, které soubory byly infikovány a ponechat tak na uživateli, aby se rozhodl, který z infikovaných souborů chce odstranit a který ne. Také by bylo vhodné zobrazit i dodatečné informace o skenovaném USB uložišti, jako je jeho velikost, název atp. K tomu všemu bude zapotřebí výstup pro obraz. Na trhu je dostupné velké množství displejů, které lze k tomu využít. HDMI konektor nabízí připojení i běžného monitoru k osobnímu počítači, to je ale pro přenosnost zařízení nevhodný výběr. Proto bude využito sedmi palcového dotykového displeje od Raspberry.

Poslední důležitou věcí je výběr programovacího jazyka. Na výběr se nabízí opravdu nepřeberné množství, a tak výběr omezuje pouze použitý operační systém. Pokud pro vybraný operační systém je dostupný interpret nebo překladač pro daný jazyk, lze jej využít pro implementaci aplikace. Nejvíce dostupných operačních systémů pro Raspberry Pi je založeno na linuxovém jádře. Na Linuxu lze využít téměř všechny existující programovací jazyky. Je proto vhodné si zvolit jazyk takový, který usnadní co nejvíce práce při implementaci. Jelikož bude v práci potřeba práce se systémovými knihovnami, je nejvhodnější zvolit jazyk C++ nebo třeba Python, protože obsahují spoustu nástaveb pro práci s těmito systémovými knihovnami. V téhle práci se dále bude využívat jazyk Python.

6.2 Python

V následující sekci bude krátký úvod o tomhle jazyku a knihovnách použitý k vypracování práce. Lehce budou také nastíněny základní konstrukce jazyka. Python[2] vznikl v roce 1991 a byl vytvořen Guido van Rossumem. Podařilo se mu vytvořit interpretovaný vysokoúrovňový programovací jazyk, který si získal širokou základnu uživatelů. Jazyk je celkově zaměřen na jednoduchost kódu a syntax, která programátorům umožní psát složitější algoritmy na pár řádků.

Pro oddělování bloků kódu nepoužívá jako většina jazyků závorky ale odsazování pomocí bílých znaků. Je třeba si proto zvolit jednotný systém odsazování a ten používat v celém programu, jinak se program nepodaří interpretovat a interpretace tak skončí s chybovou hláškou. Další výraznou vlastností je systém dynamického typování, který zjišťuje typ proměnných až za samotného běhu programu. Jedná se o takovou dvousečnou zbraň. Pokud jazyk využívá zběhlý programátor, který má přehled, co se v jeho kódu zrovna odehrává, nebude mít s dynamickým typováním sebemenší problémy. Díky tomu dochází ke ztrátě přehledu o tom, která proměnná je jakého datového typu, a proto je zapotřebí volit vhodné názvy proměnných a dávat pozor, co se do proměnné přiřazuje. Co se týče uvolňování nevyužívané paměti, o tu se stará automatická správa paměti, které se říká **Garbage collector**. Python podporuje několik programovacích paradigmat, mezi které patří objektově orientované, imperativní, funkcionální a procedurální paradigmata. Velikou výhodou jazyka jsou dostupné knihovny, které ušetřují rozsáhlé množství práce při programování. Jazyk obsahuje rozsáhlou standardní knihovnu, která je dále doplněna řadou knihoven vytvořených externě. Jedna z takových externích knihoven bude zapotřebí k implementaci, a proto bude popsána v následující podsekci.

6.2.1 Pyudev

Jedná se tedy o knihovnu, která je nedílnou součástí práce, a tím je **pyudev**[3]. Jelikož je cílem práce vytvořit automatický detektor virů na USB uložištích, bude zapotřebí detekovat připojení a odpojení takového USB zařízení, aby byl možný celý proces automatizovat bez zásahu uživatele. O správu v téhle oblasti se na linuxovém jádru stará podsystém nazývaný **udev**, resp. knihovna libudev. Aby bylo možné s touhle knihovnou pracovat přes programovací jazyk Python, je zapotřebí využít například právě knihovny pyudev, která takovou nástavbu zajišťuje. Knihovnu je možné stáhnout z GITu, anebo se nabízí možnost využití správce balíčků pro Python **pip**. Jedním příkazem lze poté knihovnu stáhnout a nainstalovat a to pomocí *pip3 install pyudev*.

Mezi hlavní funkce, které knihovna nabízí, spadá výčet připojených zařízení (nejen USB), které je možné filtrovat pomocí specifických kritérií. Pomocí systému dotazů lze

o zařízení získat různé informace, vlastnosti a atributy. Tohle všechno lze využít k zobrazení informací o zařízení, ale hlavní částí je právě monitorování USB zařízení. Existují dva způsoby. Monitorování je možné provádět jak synchronně, tak asynchronně pomocí vláken běžících na pozadí. Tyhle monitorovací vlákna zajistí odchytávání signálů, které signalizují připojení nebo odpojení USB zařízení. K dosažení takové funkce lze pomocí knihovny docílit pomocí následujících řádků kódu:

```
def activeScanningForUSBDevices():
        context = Context()
        monitor = Monitor.from_netlink(context)
        monitor.filter_by(subsystem='block')
        for device in iter(monitor.poll, None):
                if device.action == 'add':
                        #do stuff
                if device.action == 'remove':
                        #do stuff

_thread.start_new_thread(activeScanningForUSBDevices, ())
```

Pro získání informací o zařízeních je zapotřebí nejdříve vytvořit spojení s udev databází, která všechny informace obsahuje. K tomu slouží právě třída *Context*. Usnadňuje veškerou práci, protože již obsahuje implementované různé metody, pomocí kterých lze zjišťovat o připojených zařízeních potřebné informace. Aby bylo možné sledovat připojení nebo odpojení USB zařízení v reálném čase, je zapotřebí něco, co bude tyhle událost hlídat. Jelikož je knihovna pyudev přesně na tyhle věci stavěná, nabízí se třída *Monitor*. Na třetím řádku výše uvedeného kódu dochází k připojení zdroje událostí k monitoru, které má odchytávat. K tomu poslouží předem vytvořená instance třídy *Context*. Dále se nabízí možnost pomocí metody *filter_by()* nastavit u monitoru filtrování podsystémů, aby monitor zbytečně nezachytával nepotřebné události. Následuje nekonečný cyklus přes zachycující události a jednoduchým způsobem rozvětvení pro obsluhu jednotlivých situací, které můžou nastat. V případě téhle práce bude zapotřebí po detekci připojení USB zařízení provést *mount* do předem zvolené složky, následné spuštění automatického skenování proti malwaru a vyhodnocení výsledků. Jelikož se jedná o aktivní monitorování, je zapotřebí vytvořit vlastní vlákno, které ho bude obsluhovat. Jinak by docházelo k problémům při obsluze grafického rozhraní, kde by s největší pravděpodobností docházelo k známé hlášce „Application not respoding“.

Jako alternativa ke knihovně pyudev se nabízí k použití například knihovna **pyusb**. Knihovna by měla nabídnout podobné funkce jako pyudev, avšak je navíc funkční nejen na operačním systému Linux ale i Windows. Funkčnost by měla být zajištěna i na jiných platformách, pokud splňují určitá kritéria.

6.3 Grafické uživatelské rozhraní

Aby bylo možné vytvořit uživatelské rozhraní, bude zapotřebí využít nějaký framework, který tuhle funkci umožňuje. Pro jazyk Python existuje poměrně rozsáhlé množství dostupných frameworků, které jsou buďto zaměřeny přímo na jednu platformu, anebo jsou tzv. multiplatformní. Jedním z nejznámějších je pravděpodobně **Tkinter**[6], který bývá součástí balíčků, které jsou nainstalovány zároveň s interpretem pro Python. Framework je v dnešní době trošku zastaralý, ale přesto pomocí něj lze dokázat vytvořit bez problémů

plnohodnotnou aplikaci. Jedná se o poměrně jednoduchý framework se kterým se lze velmi rychle naučit pracovat.[24]

Za jako další poměrně známý framework lze označit **wxWidgets**[7], který je dostupný zdarma k jeho použití. Knihovna je napsána v jazyce C++ a místo napodobování grafických prvků používá nativní grafické prvky na podporovaných platformách. Stabilní verze je otestována na platformách Microsoft Windows, Mac OS X, Linux/Unix, OpenVMS (serverový systém) a OS/2 (IMB). Verze pro vestavěné systémy zatím není dostupná, ale podle nejnovějších zpráv by měla být ve vývoji. I když je knihovna implementována v jazyce C++, tak pro mnoho jazyků existují nástavby, které umožňují tuhle knihovnu používat. Mezi ně patří například C#, Erlang, Haskell, Perl, Ruby, Java, Javascript, ale právě i Python.[7]

Existuje samozřejmě daleko více použitelných frameworků a opět záleží, pro jakou platformu je zaměřena vyvíjená aplikace. Podle toho je pak vhodné zvolit správný framework. V téhle práci nebudou rozebrány všechny možné použitelné frameworky, ale za zmínku stojí alespoň Kivy, Pyforms, PyGObject, PyGUI, libavg a **PyQt**, který bude využit k implementaci a popsán níže.[24]

6.3.1 Qt a PyQt

Základem je knihovna Qt[5], která je opět napsaná v programovacím jazyce C++. Jedná se o multiplatformní aplikační framework, který je široce používán na vývoj aplikací s grafickým rozhraním. O její vývoj se postarala společnost Trolltech v roce 1999, která ji později prodala firmě Nokia. Pokud je vytvářená aplikace open source, lze framework využít bez poplatků. Pro komerční použití je zapotřebí zakoupit licenci. Jako většina frameworků i knihovna Qt obsahuje nástavby do jiných programovacích jazyků. Jedná se téměř o podobný seznam jazyků jako u wxWidgets. Jednou z nějvětších výhod Qt knihovny je rozsáhlá a přehledně zpracovaná dokumentace, ale také její vývojové programy Qt Creator a Qt Designer. Za pomocí těchto vývojových programů je možnost vytvářet grafické rozhraní pro jakýkoliv programovací jazyk. Díky tomu se lze vyhnout nepřehlednému vývoji grafiky přes kód, a tak je grafické rozhraní vytvořené v přehledném grafickém prostředí. Výsledné popisy jednotlivých oken aplikace jsou ve formátu XML a pro převedení do cílového jazyka existují většinou nástroje. V případě jazyka Python se jedná o nástroj **pyuic5**, který je součástí balíčků **pyqt-dev-tools**. Balíček lze stáhnout pomocí příkazu *sudo apt-get install pyqt-dev-tools*. Stejně jako wxWidgets používá Qt nativní vzhled, kde se vzhled aplikace vždy adaptuje na příslušnou platformu. Co se však týče podporovaných platforem, je jich daleko více než u konkurentních frameworků. Mezi desktopové platformy patří klasicky Microsoft Windows, Linux/X11, macOS. Qt také podporuje i mobilní platformy jako Android či iOS 6 a novější.[38]

Pro tvorbu grafického rozhraní se Qt jeví jako správná volba, jelikož má širokou platformní podporu a jeho využití je ve všech jazycích obdobné. Nenabízí však pouze tvorbu grafických rozhraní, ale zahrnuje i abstrakci síťových schránek, vláken, regulérních výrazů, SQL databází, SVG, OpenGL, XML a plně funkční webový prohlížeč.

Pro využití knihovny Qt pod jazykem Python je třeba využít nástavby **PyQt**[1]. Knihovna není nainstalována mezi základními knihovnami při instalaci interpretu pro Python, ale je zapotřebí ji nainstalovat externě, a to zcela bezplatně. PyQt však v sobě neobsahuje knihovnu Qt. Ta je zapotřebí opět stáhnout odděleně. Co se však týče podmínek použití PyQt, jsou stejné jako u Qt. Při vzniku téhle práce byly dostupné dvě verze knihovny. Jedná se o PyQt 4, která pracuje s Qt4 i Qt5 a PyQt 5, která již podporuje pouze Qt5. Společnost vyvíjející Qt již zrušila podporu pro Qt4, a proto je vhodnější využít její novější verzi 5.

Pro představu, jak jednoduché je využití knihovny, je níže uveden krátký kód pro vytvoření hlavního okna aplikace.[41]

```
from PyQt5.QtGui import *
app = QApplication(sys.argv)
form = QMainWindow()
form.show()
sys.exit(app.exec_())
```

Nejdůležitější je naimportování knihovny, abychom mohli využívat její třídy. Důležité je vytvoření aplikační objektu, který má na starosti běh celé aplikace. Pro zobrazení okna aplikace, je zapotřebí nejdříve vytvořit instanci nějaké třídy. K tomu lze využít například třídu *QMainWindow*, kde již název vypovídá o tom, že se jedná o hlavní okno aplikace. To znamená, že takové okno bude obsahovat lištu a panel nástrojů. Vytvoření samotné instance třídy však nic neznamená. U vytvoření instance lze nastavit nejrůznější atributy, které ovlivňují vzhled. Pro všechno jsou již implementovány metody, o kterých se lze dočíst v dokumentaci. Důležitá je však metoda *show()*, která zajišťuje zobrazení okna aplikace. Opačným způsobem funguje metoda *close()*. Poslední řádek v kódu slouží ke vstupu do smyčky událostí aplikace, aby nedošlo k okamžitému ukončení aplikace po jejím spuštění. Mimo QMainWindow se dají využít i třídy *QWidget* nebo *QDialog*, které mají odlišnosti ve svém využití. QWidget je však základní třídou všech objektů uživatelského rozhraní.

6.4 Výběr operačního systému

V kapitole, kde bylo rozebíráno Raspberry Pi, již byly zmíněny některé dostupné operační systémy, které by mohli být využity pro zhotovení práce. Výběr operačního systému je vhodné vybírat podle toho, co bude vyvíjeno za aplikaci. Pro zhotovení téhle práce bude stačit oficiálně uznávaný systém pro Raspberry, kterým je Raspbian. Pro osobní počítače je dostupný v x86-64 verzi a pro Raspberry samozřejmě ve verzi pro ARM.

Pokud by měla být aplikace zaměřena na osobní počítače, je vhodnější zvolit verzi x86-32. Při použití Raspbianu, který je x84-64, nastaly různé problémy při instalaci antivirových programů a debian balíčků, které jsou potřebné právě pro jejich chod. Oficiální stránky pro Raspberry nabízejí množství dalších systémů třetích stran. Z nabízených systémů na stránkách stojí za zmínku systém z odnože Ubuntu, a tím je Ubuntu Mate. Ke stažení jsou dostupné verzi jak x86-64, tak x86-32 ale i ARM pro Raspberry.

Pokud by aplikace měla být zaměřena na osobní počítače, bylo by vhodnější zvolit verzi x86-32 verzi, jak již bylo zmíněno. Je to hlavně z toho důvodu, že pro tuhle verzi je více dostupných antivirových programů. Tahle práce se však zaměřuje na platformu Raspberry Pi, a tak pro vývoj byl zvolen systém Raspbian.

6.4.1 Rozdílné architektury

Vývoj aplikace pro Raspberry většinou probíhá na stolním počítači, aby nedocházelo ke zbytečnému omezování rychlostí hardwaru. Nejjednodušším způsobem je stažení obrazu systému, který bude následně použit i na Raspberry. Tím však může nastat problém, že ve virtuálním prostředí se podaří na systém nainstalovat i programy, které se později na Raspberry nainstalovat nepodaří. Je zapotřebí si nejdříve ověřit, zdali požadované programy jsou dostupní i pro architekturu ARM. Tím se dá vyhnout nepříjemnému zjištění nefunkčnosti aplikace po jejím přenosu ze stolního počítače na Rasperry. Tenhle problém je

způsoben tím, že různé architektury využívají různé instrukční sady pro svůj procesor. Pro úspěšné nainstalování a spuštění aplikace je zapotřebí verze, která byla přeložena právě pro instrukční sadu na procesorech ARM. Přeložení aplikace bez dostupnosti zdrojových kódu nemůže zajistit sám uživatel, ale je to plně na výrobci aplikace.

Aby k těmto problémům nedošlo, lze využít například emulaci systému. Existují dvě možnosti. Jednou z nich je emulovat ARM systém na stolním počítači, čímž se zajistí, že nebude potřeba neustále testovat dostupnost programů pro ARM, nebo je možnost využít emulace systému s architekturou x86 nebo x86-64 přímo na Raspberry. Tím je umožněno použití téměř veškerých prostředků, které jsou pro tyto platformy dostupné. K dosažení emulace se nabízí využití programu **Qemu**[4]. Problém poté ale nastává při výběru operačního systému. Emulovat lze ovšem kterýkoliv systém, ale ne pro všechny jsou na internetu dostupné potřebné soubory pro emulaci. K připravení takových souborů je potřeba hlubších znalostí z téhle oblasti. Pro Raspberry jsou k dostání na internetu připravené obrazy disku s Raspbianem, kde je již přeindstalovaný emulátor pro desktopový Raspbian s architekturou x86-64, anebo právě Ubuntu Mate x86-32. Další z možností je poté využití placeného programu **ExaGear**[21] od firmy Eltechs, který se postará o veškerou práci.

Emulování systému však není úplně nejvhodnější způsob pro implementaci aplikací. Proces emulace je poměrně náročný a tím připraví zařízení o značnou část výkonu. V článku o emulaci právě přes Qemu je uvedeno, že díky probíhající emulaci je výkon procesoru snížen na rychlost 300Mhz procesoru Pentium[11]. Což je podstatná změna oproti dostupnému čtyřjádrovému procesoru s frekvencí 1,2GHz. Ani využití emulace nemusí zajistit to, že se podaří na zařízení nainstalovat všechny potřebné prostředky.

6.5 Použité antivirové programy

Poslední nedílnou součástí aplikace je výběr antivirového programu. I v téhle oblasti nastává menší problém a tím je nízká dostupnost antivirových programů pro linuxové systémy. Původně firmy vyvíjely své programy pro co nejvíce platforem. Postupem času se však přišlo na to, že na linuxové systémy není směrováno příliš mnoho útoků, a tak je trh příliš malý, aby pro něj byly udržovány verze antivirů. Podporu linuxu většina firem proto zrušila a zůstalo při ní pouze pár vybraných firem. Ty se však zaměřují především na serverové antivirové programy, které mají za úkol skenovat spíše síťový provoz, elektronickou poštu atp. Nejčastěji se všude uvádí, že Linux žádný antivirový program nepotřebuje, protože na něj nejsou útoky cíleny. Největší hrozbou pro Linux jsou především rootkity, na které však existuje dostatečné množství nástrojů.

V kapitole, kde se rozebíraly dostupné antivirové programy, již byly vybírány programy takové, které původní podporu pro Linux měli anebo ještě stále mají. Důležitou vlastností pro tuhle práci je především to, aby se u antiviru nacházela možnost jeho ovládání pomocí příkazové řádky. Bez téhle možnosti by se těžce vyvíjel automatizovaný skript. Podstatná část je i to, na jaký malware si antivir zaměřuje. Skenování elektronické pošty nebo rootkitů by v tomhle případě nemělo sebemenší význam.

Nejvěrnějším je právě první z uvedených antivirů, **ClamAV**. Tenhle antivir se bez větších potíží podaří nainstalovat na jakoukoliv distribuci Linuxu a splňuje všechny výše uvedené požadavky. Jeho nainstalování je pro běžné uživatele Linuxu jednoduchou záležitostí. Pro uživatele, kteří nemají s Linuxem zkušenosti, není instalace o moc složitější, jelikož je na internetu spousta dostupných diskuzních fór, kde je většina problémů již dávno vyřešena. K nainstalování tak stačí napsat příkaz *sudo apt-get install clamav* do terminálu. Po proběhnutí instalace lze spustit skenování, kterému se říká on-demand, pomocí příkazu *sudo*

clamscan -i -r /folder_path. Přepínač -i říká, že na výstupu budou vypsány pouze infikované soubory. Bez použití tohoto přepínače by byly na výstupu uvedeny všechny proskenované soubory. Aby bylo docíleno skenování i v podsložkách, je zapotřebí povolit rekurzivní skenování pomocí přepínače -r. Problém tohoto antiviru je ten, že při skenování souboru využívá dostupnou operační paměť a pokud se mu nepovede načíst celý soubor do paměti, vypíše chybovou hlášku a soubor přeskočí.

Nyní opět přichází problém rozdílných architektur, který brání v nainstalování dalších antivirových programů, a tak jediným možným antivirem pro Raspbian a ARM architekturu je právě ClamAV. Pro využití jiných antivirů by bylo zapotřebí využít výše zmíněné emulace systému, anebo zvolit kompletně jinou platformu od Raspberry Pi, která není postavena na ARM architektuře. Pokud by se jednalo o využití aplikace pro běžný stolní počítač s x86 procesorem, lze nainstalovat i jiné dostupné antiviry. Práce obsahuje i takovou verzi, kde se podařilo bez sebemenších problémů zprovoznit antiviry Sophos, Comodo a F-Prot, avšak pro x86-32 architekturu a přesněji pod systémem Ubuntu Mate. Jiné antiviry, než je ClamAV, se nenachází v dostupné knihovně pro *apt-get*, takže je zapotřebí provést jejich stažení z oficiálních stránek a poté nainstalovat. Většinou se jedná o instalační balíčky *.deb*, které po spuštění provedou automatickou instalaci. Některé antiviry obsahují přiložený skript, který je zapotřebí spustit pro úplné nainstalování a nastavení systémů pro následné využití programu. V takovém případě bývají k instalaci přiložené textové soubory, které obsahují informace k instalaci programu.

6.6 Nastavení operačního systému

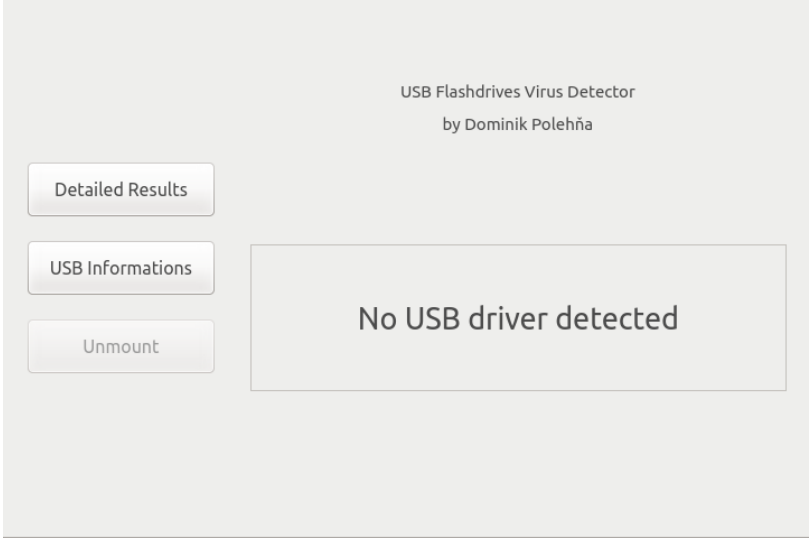
Aby bylo možné celý proces lépe automatizovat, je vhodné v systému vypnout automatické připojení (mount) USB zařízení a také automatické otevření složky s obsahem USB zařízení po připojení. Tohle všechno lze nastavit například pomocí **dconf-editor**. Editor však nebývá uvykle součástí systémů a je zapotřebí ho zvlášť nainstalovat pomocí příkazu *sudo apt-get install dconf-editor*. Pomocí editoru je možné nastavit různé věci, důležitá je však cesta **org.gnome.desktop.media-handling**, kde se nachází právě **automount**. Jeho hodnotu je zapotřebí nastavit na **false**, čímž se zabrání automatickému připojení. Aby bylo zabráněno i automatickému otevření složky s obsahem, hodnota **false** se musí nastavit i pro **automount-open**. U některých systémů může dojít k tomu, že tohle nastavení nebude stačit a je zapotřebí provést nastavení ještě na jiném místě. Cesta k druhému místu se však liší systém od systému, a proto je vhodné si tuhle informaci ověřit na internetu. Pokud není možnost stažení programu **dconf-editor**, lze tohle nastavení provést pomocí příkazové řádky zadáním příkazu *gsettings set org.gnome.desktop.media-handling.automount false*. Obdobný příkaz je i pro automount-open, pouze se změní cesta.

V systému by šlo nastavit, který program se má spustit při detekci připojení USB zařízení a tím by mohl být spuštěn například detektor virů. Výsledná aplikace téhle práce však funguje jiným způsobem a je zapotřebí, aby byla spuštěna ještě před připojením USB zařízení. Proto by bylo vhodné nastavit automatické spuštění skriptu se spuštěním systému. Existuje více způsobů, jak toho dosáhnout. Jedním z nich je, že se pomocí příkazu *sudo nano /etc/xdg/lxsession/LXDE-pi/autostart* provede otevření souboru autostart, kde lze na konec souboru připsat řádek *@xlterminal -e python3 script.py*. Pokud je v systému povoleno automatické spouštění, tak po nastavení výše uvedeného, proběhne spuštění aplikace při startu systému. Jestliže se aplikace při startu nespustí, je pravděpodobné, že je automatické spouštění aplikace zakázáno, a proto je potřeba jej povolit. Jednoduchým příkazem *sudo*

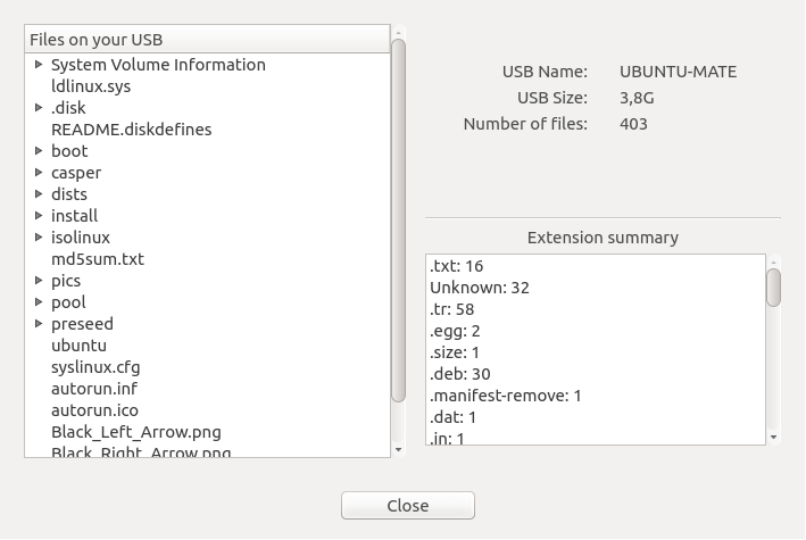
raspi-config se provede otevření konfiguračního nástroje, kde je možné tohle automatické spouštění povolit. Krokový návod k celkovému nastavení je uveden v příloze A.

6.7 Ovládání výsledné aplikace

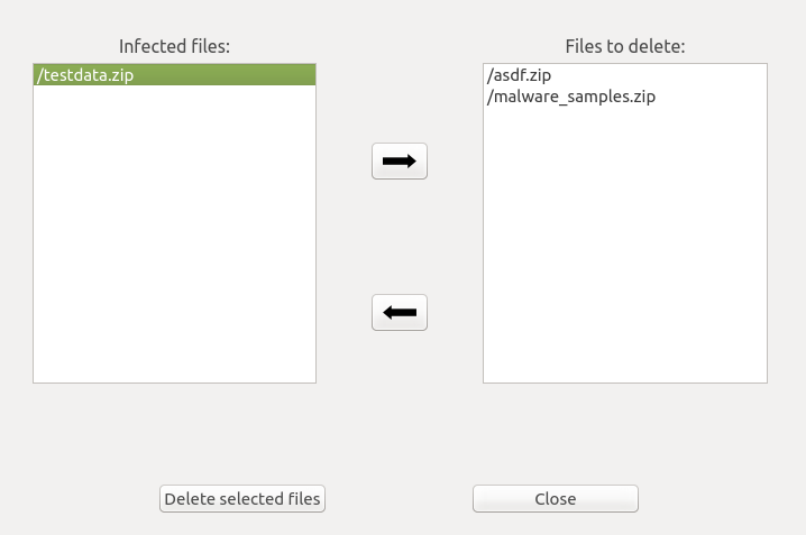
Dle představ programátora došlo k vytvoření jednoduchého grafického rozhraní, které umožňuje využít všechny funkce aplikace. Výsledná aplikace by měla fungovat následujícím způsobem, aby bylo dosaženo požadavků. Pokud dojde k připojení napájení k Raspberry Pi, automaticky se po načtení systému spustí aplikace pro detekci virů. Po spuštění se nachází ve stavu, kdy je aktivní monitorování připojených USB zařízení na hlavním okně viz obrázek 6.1. Jakmile dojde k detekci zařízení, automaticky dojde k zahájení skenování pomocí antiviru (Na verzi pro architektury x86-32 lze předem vybrat, které antiviry budou použity.). V průběhu skenování je pro uživatele již zpřístupněno okno, které obsahuje některé základní informace o skenovaném USB zařízení a je vyobrazeno na obrázku 6.2. Po dokončení skenování dojde ke zbarvení obdélníku s textem na hlavním okně, kde je pomocí barvy signalizován výsledek. Jakmile je skenování u konce a obdélník je zbarven, zpřístupní se okno s podrobnějšími výsledky, kde jsou případně zobrazeny soubory, které byly označeny za infikované. Pomocí šipek lze vybrat, který ze souborů bude odstraněn a který ponechán. Odstranění lze poté provést pomocí tlačítka pod výběrem. Ilustrační obrázek okna s podrobnějšími výsledky lze najít na obrázku 6.3. Před odpojením USB zařízení je doporučeno použít tlačítka *Unmount* na hlavním okně, aby nedošlo k poškození zařízení nebo ke ztrátě dat. Na hlavním okně, opět v obdélníku s textem, je upozornění, kdy je zařízení připraveno k odpojení. Po odpojení zařízení je aplikace připravena celou akci opakovat znovu.



Obrázek 6.1: Hlavní okno



Obrázek 6.2: Vedlejší okno s informacemi o USB zařízení



Obrázek 6.3: Vedlejší okno s výsledky skenování

Kapitola 7

Závěr

Při postupné analýze dostupných prostředků pro tvorbu práce a následné implementaci nastaly různé problémy, které stály v cestě. Přestože výsledná aplikace plní svůj účel a splňuje původní požadavky, nejedná se o úplně optimální řešení, a proto by bylo vhodné zmínit se o návrhu jistých změn. Přestože je Raspberry Pi kvalitně zpracovaný jednodeskový počítač a dobře finančně dostupný, omezuje výslednou aplikace právě díky své ARM architektuře, na které je založen. Právě architektura vytváří omezení při výběru antivirových programů. Avšak i při zvolení jiné architektury by docházelo k omezení ze strany operačního systému Linux. Pokud by se pro vývoj použil jednodeskový počítač, který by využíval x86-32 nebo x86-64 architektury, bylo by umožněno využít i jiné operační systémy. Návrh alternativních jednodeskových počítačů byl uveden v kapitole implementace v sekci o vývojovém prostředí. Při zvolení některé z uvedených platforem by došlo k razantní navýšení ceny. Jestliže by byl zvolen pro vývoj například systém Windows, došlo by ke ztrátě dostupnosti vestavěných systémových programů pro Linux, které umožňují zjednodušení práce a získání mnoho potřebných informací, ale došlo by k rozšíření množiny možných použitelných antivirů. Pro systém Windows u většiny antivirů ale není dostupné ovládání z příkazové řádky, které je pro zhotovení práce nezbytné.

Celkově všechna omezení, na určitou množinu antivirových programů, mají na svědomí nemožnost dosažení lepších výsledků při skenování. Antivirový program ClamAV, který byl využit v téhle práci, sice nabízí kvalitní bezpečnostní ochranu, ale díky své rychlosti skenování není úplně nejvhodnějším výběrem. V případě využití tohoto antiviru, na platformě Raspberry Pi, dochází k omezení ze strany velikosti operační paměti. Pokud antivir narazí na soubor, který se mu nepodaří načíst celý do operační paměti, tak dojde k přeskočení skenování tohoto souboru. To znamená, že v případě, kdy by infikovaný soubor přesahoval velikost operační paměti, nedojde k jeho odhalení. Co se týče rychlosti skenování, tak antivir ClamAV nevyužívá veškerý dostupný výkon procesoru a je zpomalován pouze svou implementací. To znamená, že výkonnější jednodeskový počítač by neměl žádný vliv na rychlost dokončení skenování. Celkově se Raspberry Pi jeví pro práci jako vhodné řešení, avšak ne úplně optimální. Vše záleží na tom, jaké požadavky má uživatel na aplikaci.

Slovník

boot-time skenování kontroluje malware ještě před zavedením operačního systému a ostatních aplikací. 16

drag and drop znamená v překladu táhni a pust. V našem případě můžeme chytit soubor, který přetáhnem na antivirus a on nám jej zkontroluje. 14

integrovaný obvod je specifikace, která definuje softwarové rozhraní mezi operačním systémem a firmwarem použitého hardwaru plural. 20

NFS je internetový protokol používání ke vzdálenému přístupu k souborům přes síť. 13

on-access jedná se o anglické slovíčko pro výraz „při přístupu“, v kontextu práce používáno pro skenování, které je spuštěno při přístupu k souboru. 12

on-demand jedná se o anglické slovíčko pro výraz „na vyžádání“, v kontextu práce používáno pro skenování na žádost uživatele. 13, 17, 18, 28

promiskuitní režim ‚jedná se o speciální režim síťové karty, ve kterém zle zachytávat síťovou komunikaci, která není přímo určená pro naše zařízení nebo počítač. 12

real-time výraz se v rámci antivirových programů používá pro skenování, které probíhá na pozadí bez našeho vědomí při používání počítače. 17, 18

RSA je šifra s veřejným klíčem. Algoritmus je vhodný jak pro podepisování, tak šifrování.. 6

shell script Jedná se o skript napsaný pro shell(program, který vytváří uživatelské rozhraní, umožňuje využívat funkce jádra operačního systému atp.). Nejčastější operace jsou manipulace se soubory, provádění programování a tisk textu.. 12

UEFI je specifikace, která definuje softwarové rozhraní mezi operačním systémem a firmwarem použitého hardwaru. 18

Literatura

[1] PyQt.
URL https://wiki.python.org/moin/PyQt

[2] Python.
URL https://www.python.org/

[3] pyudev.
URL https://pyudev.readthedocs.io/en/latest/

[4] Qemu.
URL https://www.qemu.org/

[5] Qt.
URL https://www.qt.io/

[6] TkInter.
URL https://wiki.python.org/moin/TkInter

[7] wxWidgets. Online, převzato Duben 16, 2018.
URL https://www.wxwidgets.org/

[8] About ESET Company. 2018, převzato Březen 24, 2018.
URL https://www.eset.com/int/about/

[9] A. Young, M. Y.: Cryptovirology: extortion-based security threats and countermeasures. Oakland, CA, USA, USA: IEEE, Květen 1996, iSBN 0-8186-7417-2.

[10] AG, G. D. S.: G Data presents first Antivirus solution in 1987. Online, Březen 2017, převzato Březen 24, 2018.
URL https://www.gdatasoftware.com/about-g-data/company-profile

[11] AlbrechtL: RPi-QEMU-x86-wine SD-card image. Převzato Duben 26, 2018.
URL https://github.com/AlbrechtL/RPi-QEMU-x86-wine

[12] Antivirus, T.: F-Prot Review. Převzato Prosince 2, 2017.
URL https://www.top10antivirussoftware.com/reviews/f-prot

[13] Aycock, J.: *Computer Viruses and Malware*. Springer, 2006, ISBN 978-0-387-30236-2.

[14] Barwise, M.: What is an internet worm? Září 2010, převzato Únor 13, 2018.
URL http://www.bbc.co.uk/webwise/guides/internet-worms

[15] Beatty, R.; Dore, J.; Lambert, L.; aj.: *Inventors and Inventions*. Marshall Cavendish Corporation, 2008, ISBN 0761477675.
URL https://www.amazon.com/Inventors-Inventions-Richard-Beatty/dp/0761477675?SubscriptionId=0JYN1NVW651KCA56C102&tag=techkie-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=0761477675

[16] Brulez, N.: Ransomware: Fake Federal German Police (BKA) notice. *Securelist*, Březen 2011, převzato Únor 14, 2018.
URL https://securelist.com/ransomware-fake-federal-german-police-bka-notice/29781/

[17] Centrum, A.: Sophos Endpoint Protection. Online, převzato Listopad 28, 2017.
URL https://www.antivirovecentrum.cz/sophos/business/endpoint-protection.aspx

[18] ClamAV: About ClamAV. Online, 2017, převzato Listopad 26, 2017.
URL http://www.clamav.net/about

[19] ClamAV: About ClamAV. 2017, převzato Listopad 26, 2017.
URL https://web.archive.org/web/20100804063400/http://www.clamav.net/lang/en/about/

[20] Corporation, S.: What is the difference between viruses, worms, and Trojans? Září 2016, převzato Únor 13, 2018.
URL https://support.symantec.com/en_US/article.TECH98539.html

[21] Eltechs: ExaGear.
URL https://eltechs.com/product/exagear-desktop/

[22] ESET: Nod32 Antivirus. Online, 2017, převzato Prosinec 4, 2017.
URL https://cdn1.esetstatic.com/ESET/CZ/Produktove_listy/domacnosti/eav-2018.pdf

[23] Fisher, S.: Comodo Antivirus Software Review. Online, 2017, převzato Prosinec 1, 2017.
URL https://www.thebalance.com/comodo-antivirus-review-1356601

[24] Foundation, P. S.: Graphic User Interface FAQ. Online, převzato Duben 16, 2018.
URL https://docs.python.org/3/faq/gui.html

[25] Goldberg, I. (editor): *A Secure Environment for Untrusted Helper Applications (Confining the Wily Hacker)*, San Jose, California: University of California, Berkeley, 1996, převzato Únor 17, 2018.

[26] Henry, A.: The Difference Between Antivirus and Anti-Malware (and Which to Use). Srpen 2013, převzato Únor 16, 2018.
URL https://lifehacker.com/the-difference-between-antivirus-and-anti-malware-and-1176942277

[27] Hák, I.: *Moderní počítačové viry*. Viry.cz, 2004.

[28] Jalůvka, J.: *Moderní počítačové viry*. Computer Press, 1993.

[29] Jelínek, L.: Na vaši ochranu: Antivirový program ClamAV. Online, Prosinec 2008, převzato Listopad 26, 2017.
URL https://www.linuxexpres.cz/business/na-vasi-ochranu-antivirovy-program-clamav

[30] Kizza, J. M.: *Guide to Computer Network Security (Computer Communications and Networks)*. Springer, 2008, ISBN 9781848009165.

[31] Landwehr, C. E. A. R. B. J. P. M. W. S. C.: A taxonomy of computer program security flaws, with examples. 1993, převzato Únor 13, 2018.
URL http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA465587

[32] Liptak, B. G.: *Instrument Engineers' Handbook, Third Edition, Volume Three: Process Software and Digital Networks: Volume 1*. Taylor & Francis Inc, 2002, ISBN 9781439863442.

[33] McAfee: Rootkits, Part 1 of 3: The Growing Threat. 2006, převzato Únor 15, 2018.
URL https://web.archive.org/web/20060823090948/http://www.mcafee.com/us/local_content/white_papers/threat_center/wp_akapoor_rootkits1_en.pdf

[34] Michael Myers, S. Y.: An Introduction to Hardware-Assisted Virtual Machine (HVM) Rootkits. Srpen 2005, převzato Únor 15, 2018.
URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.90.8832

[35] Nadel, B.: Avast! Free Antivirus Review: Fast, Free and Fun. Online, Srpen 2017, převzato Prosinec 3, 2017.
URL https://www.tomsguide.com/us/avast-free-antivirus,review-2208.html

[36] Naveen, S.: Anti-virus software. Červen 2016, převzato Únor 16, 2018.
URL https://www.webroot.com/in/en/home/resources/tips/pc-security/security-what-is-anti-virus-software

[37] Nelson Murilo, K. S.-J.: chkrootkit V. 0.52. Březen 2017, převzato Listopad 27, 2017.
URL http://www.chkrootkit.org/README

[38] Qt: What is Qt. Online, převzato Duben 16, 2018.
URL https://www.qt.io/what-is-qt/

[39] RaspberryPi: Raspberry Pi Hardware. Online, převzato Únor, 23.
URL
https://www.raspberrypi.org/documentation/hardware/raspberrypi/README.md

[40] Report, S.: Monitoring Software on Your PC: Spyware, Adware, and Other Software. Technická zpráva, Federal Trade Commission, Březen 2005.
URL https://goo.gl/FhcrFi

[41] Riverbank: What is PyQt. Online, 2018, převzato Duben 16, 2018.
URL https://riverbankcomputing.com/software/pyqt/intro

[42] Rubenking, N. J.: Avast Free Antivirus 2017. Online, Únor 2017, převzato Prosinec 3, 2017.
URL https://www.pcmag.com/article2/0,2817,2471522,00.asp

[43] Rubenking, N. J.: Comodo Antivirus 10. Leden 2017, převzato Prosince 1, 2017.
URL https://www.pcmag.com/article2/0,2817,2474735,00.asp

[44] Rubenking, N. J.: ESET NOD32 Antivirus. Online, Listopad 2017, převzato Prosinec 3, 2017.
URL https://www.pcmag.com/article2/0,2817,2469847,00.asp

[45] Rubenking, N. J.: Sophos Home Free. *PCMag*, Únor 2018.
URL https://www.pcmag.com/article2/0,2817,2498877,00.asp

[46] Shadowserver: Virus Yearly Stats. 2012, převzato Listopad 26, 2017.
URL http://www.shadowserver.org/wiki/pmwiki.php/Stats/VirusYearlyStats

[47] Shadowserver: Virus180-Day Stats. 2016, převzato Listopad 26, 2016.
URL http://www.shadowserver.org/wiki/pmwiki.php/AV/Virus180-DayStats

[48] SolidRun: SolidPC Q4.
URL https://www.solid-run.com/intel-braswell-family/solidpc-q4-carrier-board/

[49] Stallings, W.; Brown, L.: *Computer Security: Principles and Practice (2nd Edition) (Stallings)*. Pearson, 2011, ISBN 978-0-13-277506-9.

[50] Szor, P.: *The Art of Computer Virus Research and Defense*. Zoner Press, 2006.

[51] editorial team, S.: Sophos Home Security Free. Online.
URL https://sophos-home.en.softonic.com/

[52] Tiernan, R.: E-mail viruses blamed as spam rises sharply. Online, Únor 2004, převzato Únor 13, 2018.
URL http://old.seattletimes.com/html/businesstechnology/2001859752_spamdoubles18.html

[53] Tulloch, M.: *Microsoft® Encyclopedia of Security*. Microsoft Press, 2003, ISBN 0735618771.
URL https://www.amazon.com/Microsoft%C2%AE-Encyclopedia-Security-Mitch-Tulloch/dp/0735618771?SubscriptionId=0JYN1NVW651KCA56C102&tag=techkie-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=0735618771

[54] Wells, J.: Virus timeline. Srpen 1996, převzato Únor 16, 2018.

[55] Wikipedia: Avast Antivirus. Online, převzato Prosinec 3, 2017.
URL https://cs.wikipedia.org/wiki/Avast_Antivirus

[56] Wikipedia: ESET NOD32 Antivirus. Online, převzato Prosinec 3, 2017.
URL https://cs.wikipedia.org/wiki/ESET_NOD32_Antivirus

[57] Wikipedia: chkrootkit. Květen 2017, převzato Listopad 27, 2017.
URL https://en.wikipedia.org/wiki/Chkrootkit

Příloha A

Zprovoznění skriptu na Raspberry Pi

1. Zprovozněte na svém Raspberry Pi systém Raspbian
2. Nainstalujte Python3
   (a) V příkazové řádce spusťte příkaz *sudo apt-get install python3*
3. Nainstalujte Qt
   (a) *sudo apt-get install qt5-default qtcreator*
   (b) *sudo apt-get install python3-pyqt5*
   (c) *sudo apt-get install pyqt5-dev-tools*
4. Nainstalujte externí knihovnu **pyudev** pro Python
   (a) *pip3 install pyudev*
   (b) Nebo můžete využít některou z uvedených metod zde https://pyudev.readthedocs.io/en/latest/
5. Nainstalujte ClamAV antivirový program
   (a) *sudo apt-get install clamav*
6. Vypněte automatické připojování USB zařízení
   (a) *gsettings set org.gnome.desktop.media-handling.automount false*
   (b) *gsettings set org.gnome.desktop.media-handling.automount-open false*

V téhle fázi by měla jít aplikace spustit pomocí příkazu *python3 main.py*. Pokud chcete docílit automatického spuštění při startu systému, pokračujte v bodech níže.

1. Nastavte do autostart skriptu, aby se spustila aplikace
   (a) *sudo nano /etc/xdg/lxsession/LXDE-pi/autostart*
   (b) Na konec skriptu připiště řádek *@xlterminal -e python3 x.py*, kde místo **x.py** použijete cestu k souboru main.py

Pro využití verze pro x86 systémy, je zapotřebí doinstalovat i zbylé tři antivirové programy Comodo, Sophos a F-Prot. Soubory pro instalaci jsou přiloženy ve složce s aplikací pro x86 systémy. Poté je důležité zkontrolovat v souboru **main.py**, že u volání antivirů jsou nastaveny správné cesty k jejich binárním souborům.